

AD-A189 633

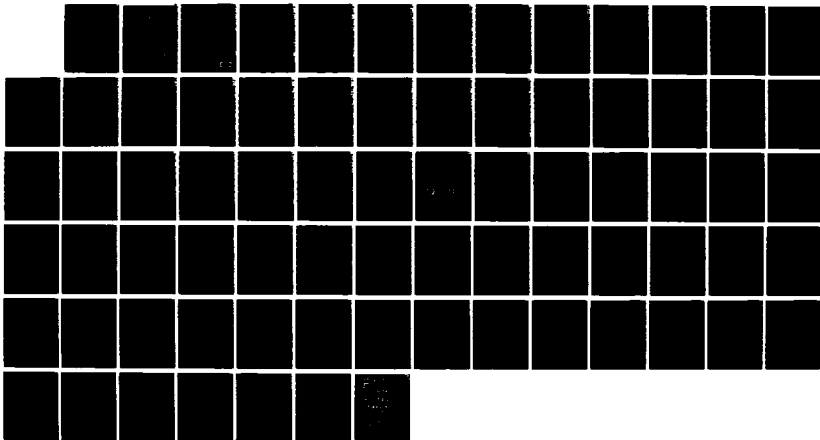
A STUDY OF THE EQUATION OF STATE FOR COPPER(U) AIR
FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF
ENGINEERING C M COURT DEC 87 AFIT/GEP/EMP/87D-4

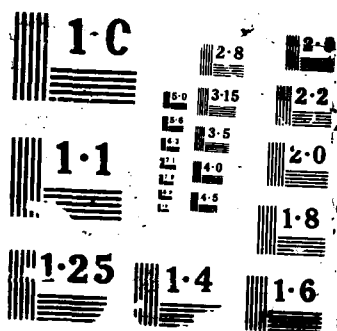
1/1

UNCLASSIFIED

F/O 12/1

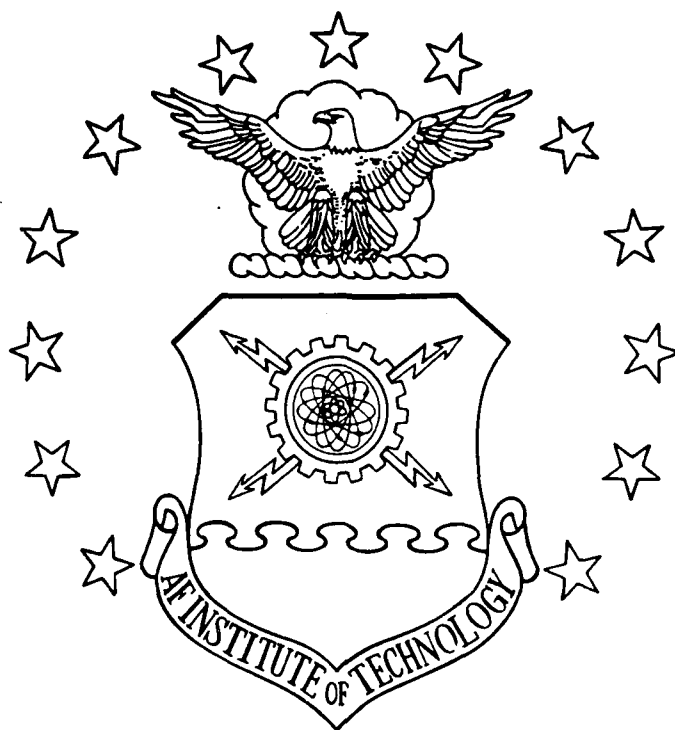
NL





DTIC FILE COPY

AD-A189 633



A STUDY OF THE EQUATION OF STATE

FOR COPPER

THESIS

Charles M. Court
Major, USAF

AFIT/GEP/ENP/87D-4

DTIC
ELECTE
MAR 03 1988
S H

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

DISTRIBUTION STATEMENT A

Approved for public release

88 3 01 050

①
AFIT/GEP/ENP/87D-4

A STUDY OF THE EQUATION OF STATE
FOR COPPER
THESIS

Charles M. Court
Major, USAF

AFIT/GEP/ENP/87D-4

Approved for public release; distribution unlimited

DTIC
ELECTE
MAR 03 1988
S a D
H

AFIT/GEP/ENP/87D-4

A STUDY OF THE EQUATION OF STATE
FOR COPPER

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Engineering Physics

Charles M. Court, B.S., B.A., M.B.A.

Major, USAF

December 1987

Approved for public release; distribution unlimited

Acknowledgements

Many people helped me during this fleeting period of thesis preparation. I give my highest praise and sincere thanks to my advisor, Capt Michael Sabochick, for his leadership and for his complete knowledge of every phase of atomistic simulation. We should have instructor pilots so able and so well versed. I also thank Major James Lupo, who corrected my programming when I badly needed help; Mr Rick Norris, who kept the parallel processing computers working despite overheats and power failures, and Lt Col James Mills, whose last minute editing comments meant the difference between clarity and nonsense. I also thank my wife, Marilyn, for believing all of this would work despite my setbacks and discouragements.

Charles M. Court



Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Avail and/or	
Dist	Special
A-1	

Table of Contents

	Page
Acknowledgements	ii
List of Figures	v
Abstract	vii
I. Introduction	1-1
General Approach	1-2
Significance of this Study	1-4
Order of Presentation	1-5
II. Atomistic Simulation	2-1
Building the Lattice	2-2
Boundary Effects	2-4
Force Calculations	2-7
The Embedded-Atom Method	2-8
Integration and Computation	2-10
Calculation of Parameters	2-11
Determining the Time Steps and Length of Runs	2-13
Alternative Configurations	2-14
Time Saving Steps	2-14
III. Parallel Processing	3-1
Parallel Processing Computers	3-1
Molecular Dynamics as a Parallel Process	3-2
The Embedded-Atom Solution in Parallel	3-5
IV. Experiments and Results	4-1
The Equation of State for Copper	4-1
Validation -- Copper at Room Temperature	4-3
Determination of the Equation of State	4-5
The First Series of Heating Runs	4-5
A Second Form of Validation	4-5
Melting After Superheating	4-7
Freezing the Melted Crystal	4-7
Melting the Crystal Again	4-18
Parallel Processing Results	4-22

V.	Conclusions and Recommendations	5-1
	Conclusions	5-1
	Recommendations	5-3
Appendix A:	Calculation of the Potential	
	Time Savings	A-1
Bibliography	B-1
Vita	C-1

List of Figures

Figure		Page
2-1.	A Face-Centered Cubic Crystal	2-3
2-2.	Periodic Boundaries	2-5
2-3.	How an Atom is Replaced if it Crosses a Periodic Boundary	2-6
2-4.	Neighbor Interaction Spheres	2-15
3-1.	Possible Node Communication Paths	3-3
3-2.	The Interaction Cutoff Range and Adjacent Lattice Boxes	3-4
3-3.	A Parallel Processing Block Diagram	3-7
4-1.	Computer Calculated and Expected Linear Expansion vs. Temperature	4-6
4-2.	Volume of the Copper Crystal as it is Melting	4-8
4-3.	Potential Energy of the Copper Crystal as it is Melting	4-9
4-4.	Linear Expansion of the Copper Crystal as it is Melting	4-10
4-5.	Volume of the Copper Crystal at the Melt Temperature	4-11
4-6.	Potential Energy of the Copper Crystal at the Melt Temperature	4-12
4-7.	Linear Expansion of the Copper Crystal at the Melt Temperature	4-13
4-8.	Volume of the Copper Crystal as it Freezes .	4-15
4-9.	Potential Energy of the Copper Crystal as it Freezes	4-16
4-10.	Linear Expansion of the Copper Crystal as it Freezes	4-17
4-11.	Volume of the Copper Crystal at the Actual Melt Point	4-19

4-12.	Potential Energy of the Copper Crystal at the Actual Melt Point	4-20
4-13.	Linear Expansion of the Copper Crystal at the Actual Melt Point	4-21
4-14.	Percentage of Time in the Force and Neighbor Subroutines vs. Number of Nodes	4-25

Abstract

The purpose of this study was to use the computer technique of atomistic simulation to investigate the equation of state for copper. Specifically, this study used a deterministic technique called molecular dynamics with a relatively new method to determine interatomic forces called the embedded-atom method. The basic approach was to modify a computer program which successfully works on the Cray supercomputer to work on other computers which are more readily available. As a final step, the program was modified to work on a parallel processing computer system in order to study the time savings made possible by solving parts of the problem in parallel.

This study found that the molecular dynamics approach and the embedded-atom method gave an accurate prediction of the copper lattice constant at room temperature and of thermal expansion at higher temperatures. In order to determine the melting temperature, the simulated crystal was then heated past its melt point, cooled until it froze, and reheated until it melted again. The experiments found a transition temperature very near 1350° K, which is close to the accepted melt point of 1357.6° K.

The experiments with the parallel processing system showed that a program running in a parallel mode could realize time savings of a factor of 3.75 over serial processing systems. Because the code was essentially modular in nature, the most extensive modifications to make it run in parallel involved separating the necessary subroutines and adding the communications routines so data could be passed back and forth during the calculations. The modifications to run in parallel did not make the program more difficult to manipulate and could quickly be removed in order to run the program on a normal, serial computer.

A STUDY OF THE EQUATION OF STATE FOR COPPER

I. INTRODUCTION

Atomistic simulation is a computer technique used to study matter. The general idea is for a computer to simulate a system of discrete atoms. As such, atomistic simulation differs from continuum methods such as fluid dynamics. The atoms are allowed to react to random motions and to interatomic forces, and the computer determines each atom's position and velocity at discrete time intervals. The computer then calculates properties such as temperature and pressure, which are functions of atomic position and velocity. These functions are averaged over several thousand time steps so that the overall state of the atomic system can be determined.

An interatomic potential, the energy between atoms as a function of distance, is a way to express how atoms interact with each other. The forces on each atom can be calculated after the total energy of a system has been calculated (15:1). Atomistic simulation has the advantage of being able to work with many different interatomic potential functions. On the other hand, the application of the results of atomistic simulation experiments depend heavily upon the accuracy of the interatomic potential function applied within the computer program.

Atomistic simulation gives the experimenter complete control of the external environment which, in turn, allows environments that may be unavailable in the laboratory. The computer can generate large quantities of data on the material studied, and extremely short-lived phenomena can be simulated, such as the motion of phonons within a crystal (4) and the melting of metallic crystals (8).

The purpose of this study was twofold. First, the equation of state for copper was investigated by using: 1) a particular atomistic simulation technique, molecular dynamics, and 2) a relatively new technique to determine interatomic force, the embedded atom method. Second, this study used both serial and parallel processing computers in an attempt to generate accurate data in the least amount of time and to validate the use of parallel processing techniques on this particular type of problem.

General Approach

This series of experiments used a computer code developed by Baskes and Foiles (1) for use on the Cray computer and modified by the researcher to run on the ELIXI System 6400 under the EMBOS 11.3 operating system. The modified code was run extensively on the ELIXI system in order to validate the output and to perform several computer experiments on both crystallized and molten copper.

The code was further modified to run on the Intel Personal Supercomputer (iPSC). The iPSC is a computer which employs up

to thirty-two concurrent processors. Each processor can work independently of the others. Ideally, any given program is solved much more quickly than on a conventional serial processor. Unfortunately, not every aspect of the molecular dynamics program lends itself to a parallel processing approach. The techniques and results are discussed later in this text.

The steps which the computer uses to solve the problem are:

- 1) establish matrices that describe the type of material, position, and velocity of every atom in the crystal to be studied in the experiment,
- 2) determine the energies and forces on each atom for the current time step,
- 3) use those forces to project atomic accelerations and displacements over the time step,
- 4) use the displacements and forces to provide snapshots of the locations of the atoms at given time intervals,
- 5) calculate state functions such as system energy, internal pressure, internal temperature, and stress, and
- 6) begin the next time step by calculating the forces and continuing with the next displacements.

When a specified number of time steps is complete, averages taken over all time steps (after a short period to let forces even out) are used to determine the final state of the original material. The final state can then be stored in a computer file and used as initial positions for subsequent computer experiments.

Significance of this Study

This study is significant in three areas: 1) as a straightforward study of solid and liquid copper, 2) as a validation of a particular embedded-atom potential for copper, and 3) as an expansion in the use of parallel processing computers.

Copper is well known for high electrical conductivity and for efficient heat diffusion. Any further study of metals in general and of copper in particular can add further insight into the nature of matter. As new alloying techniques and new methods to alter surface molecular arrangements continue to develop, extensive knowledge of material will become increasingly critical (16:227,228).

Should molecular dynamics become a more necessary scientific tool, programs must be applied to computers which are more readily available than a Cray supercomputer. During this study the modified code operated on other computers which are less costly and more accessible. This advance indicates that molecular dynamics can be used by students and other investigators more easily than before.

This study also shows that the embedded-atom method gives useful interatomic potentials. The other advantages of the embedded atom method, such as its applicability to other metals and alloys, will make this approach useful in other atomistic simulation approaches.

Finally, the Air Force is constantly attempting to increase computer control of aircraft and other weapons systems.

Parallel processing promises to be much faster than conventional serial processing, and that speed will be extremely useful in managing interdependent tasks often found in flight control, communications, and electronic warfare systems. Advances in the application and understanding of parallel processing are important for both scientific investigation and for future Air Force needs.

Order of Presentation

The specifics of molecular dynamics are detailed in Chapter Two. The discussion includes the important calculations completed during the simulation and the different methods used to determine the interatomic potentials needed for force calculations.

Chapter Three discusses parallel processing, how it differs from conventional serial processing, and how molecular dynamics and the embedded atom method can be adapted to parallel processing techniques.

Chapter Four discusses the computer experiments and their results. The ELIXI, which uses conventional serial computing, gave extensive data on the equation of state of copper. The iPSC parallel processing computer was used to show the potential time savings by using parallel computing techniques.

The last chapter, Chapter Five, discusses conclusions and makes recommendations.

II. ATOMISTIC SIMULATION

The advent of the programmable electronic computer has made atomistic simulation a feasible experimental tool. Physicists in the laboratory can only observe statistical parameters such as temperature and pressure. Although the sum of the forces on each atom determines those observable parameters, no one could observe the motions of each individual atom. Atomistic simulation makes the forces on each atom observable and the computer can use the sum of the forces on each atom in order to calculate the parameters involved in the equation of state.

The simulation technique used in this study is called molecular dynamics. Molecular dynamics is almost completely deterministic. After random numbers with a Boltzmann distribution are used to assign initial velocities to each atom, atomic motions depend entirely on interatomic potential calculations. The computer calculates subsequent atomic motions and uses those motions to determine parameters of state.

The purpose of this chapter is to discuss the mechanics of atomistic simulation. The opening section discusses how the lattice is simulated within the computer and following sections discuss subsequent steps of the solution process: calculating the interatomic forces, integrating those forces and determining the atomic motions, and calculating the parameters of state from the atomic motions. This chapter

concludes by discussing other pertinent problems such as the determination of the length and number of time steps necessary to solve a given problem, what alternative configurations can be computed from the molecular dynamics approach, and what techniques are available to save computer time.

Building the Lattice

A crystal of 256 copper atoms is used for all computer runs in this study because it provides a good compromise between the desire for a large crystal and the limitations on computer speed and capacity (13:20). Copper forms face-centered cubic (fcc) crystals and each unit cell of an fcc system contains four atoms (19:12;12:6). Figure 2-1 shows how the atoms are arranged in a single fcc crystal. A crystal of four lattice constants on each side gives ($4 \times 4 \times 4 = 64$) unit cells or ($64 \times 4 = 256$) atoms in all.

Each atom in the crystal is then described by its position and its velocity in each of the three spatial dimensions. Hence, each atom must have six coordinates:

spatial position (x, y, z) and velocity $\left(\frac{dx}{dt}, \frac{dy}{dt}, \frac{dz}{dt}\right)$

Initial positions are determined by the type of lattice structure and by the limits on crystal size. Velocities are generated by assigning random numbers with a Boltzmann distribution based on the initial temperature.

After the matrices to describe the crystal are built and used in a computer experiment, their positions and velocities can be stored in a file. Subsequent computer runs can use that

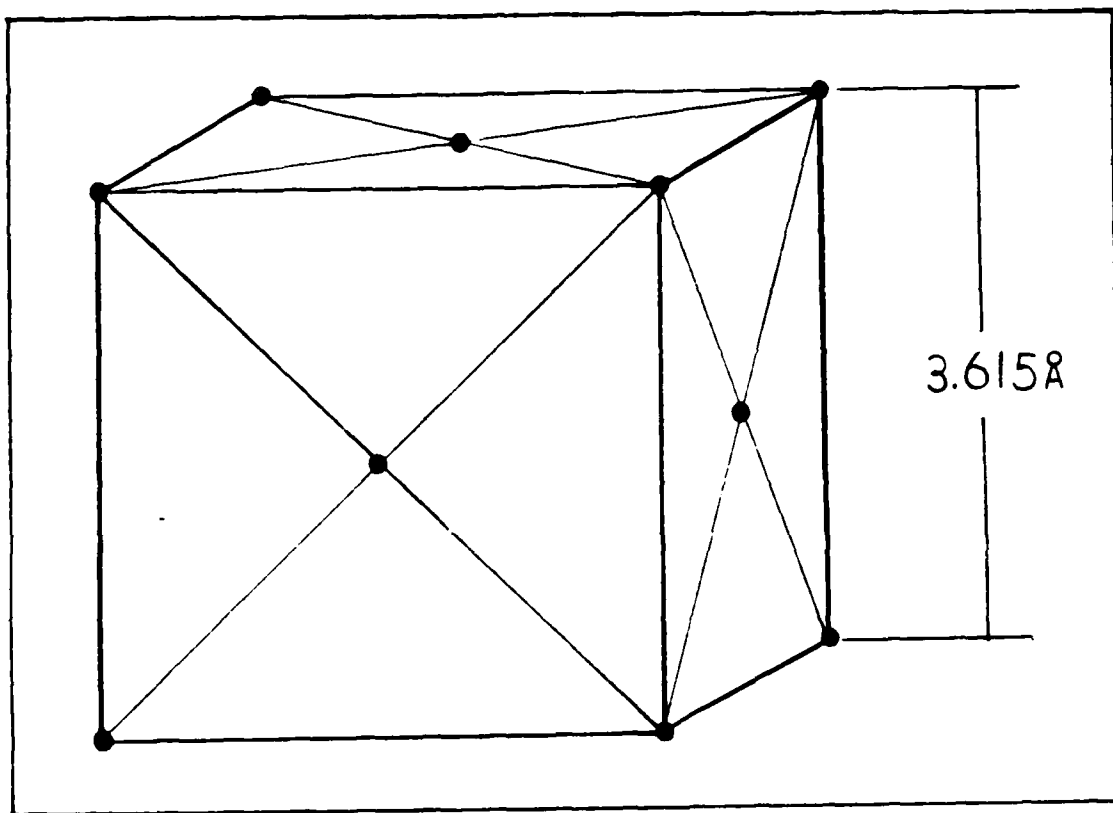


Figure 2-1. A Face-Centered Cubic Crystal
(from 12:4)

file as initial positions and velocities so the lattice does not need to be rebuilt. With this procedure, for example, a single crystal can be heated and cooled over the course of several computer runs.

Boundary Effects

In order to eliminate boundary effects, the atomic system is simulated to be surrounded by images of itself. This is illustrated on Figure 2-2. The distances between the actual crystal and its images, the periodic bounds, are specified by the program operator. If an atom crosses a periodic boundary during any time step, that atom is placed on the other side of the crystal as shown in Figure 2-3. Experience with the program demonstrated that if the operator imposes unrealistic periodic bounds, the simulation responds with unrealistic pressures and temperatures.

An alternative technique to minimize boundary effects on a single crystal would be to use a very large crystal with vacuum boundaries. Surface area, in general, tends to expand at a rate proportional to the square of any increase in the number of atoms; mass and volume increase in proportion to the cube of any increase. As a crystal becomes larger, therefore, the proportion of surface area to mass would become smaller, and surface effects would decrease. A crystal of one million atoms or more, for example, would have such minimal boundary effects. Extremely large crystals are not practical for the computers available at this time.

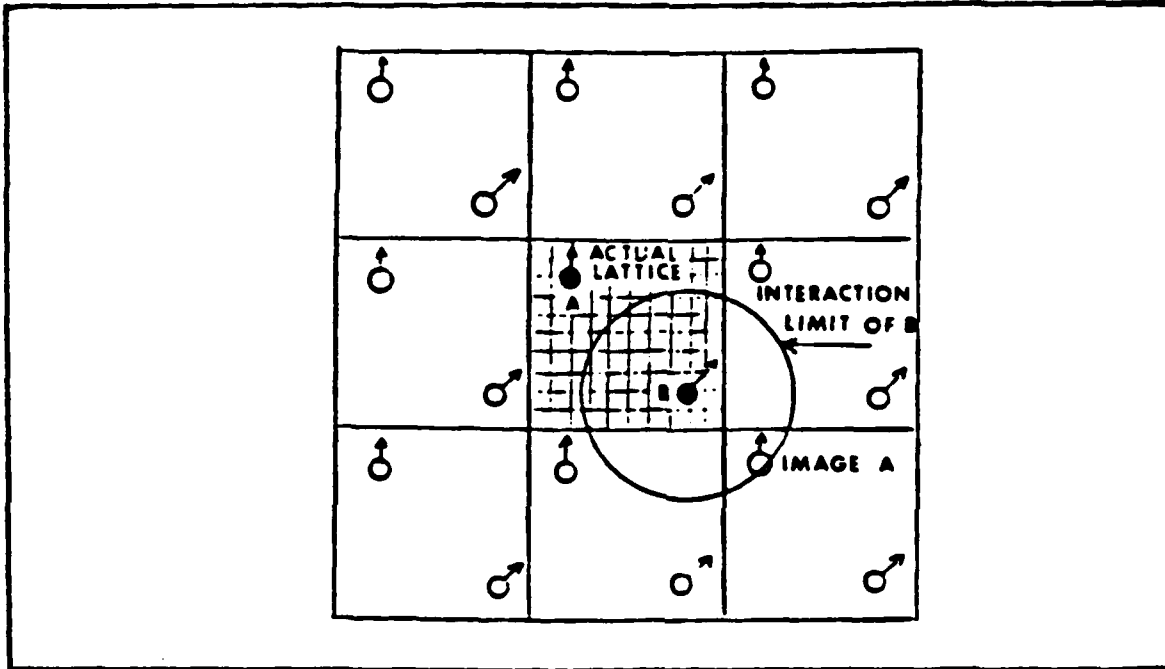


Figure 2-2. Periodic Boundaries
 B interacts with the image of A
 (arrows indicate velocity vectors)
 (from 17:11)

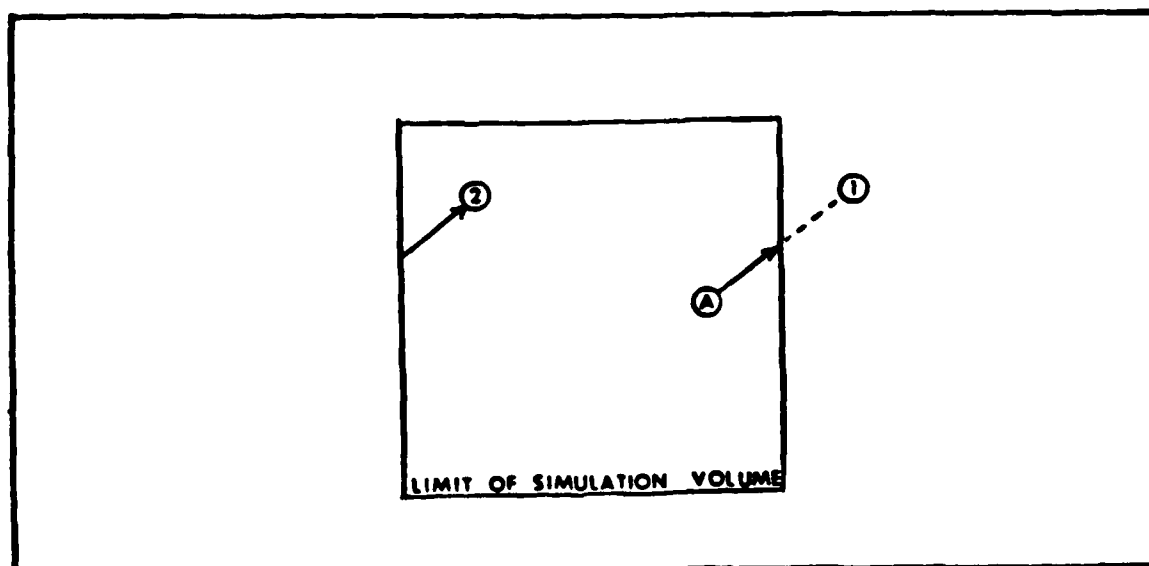


Figure 2-3. How an Atom is Replaced if it Crosses a Periodic Boundary
(From 17:12)

Force Calculations

Force and energy calculations are the core of any molecular dynamics program. The general idea is that the forces on an atomic system can be computed by summing the forces on each individual atom:

$$U(1, N) = \sum_i u_1(r_i) + \sum_{\substack{i,j \\ i < j}} u_2(r_i, r_j) + \sum_{\substack{i,j,k \\ i < j < k}} u_3(r_i, r_j, r_k) \\ + \dots + u_N(r_1, r_2, r_3, \dots, r_N) \quad (2.1)$$

where $u_1(r_i)$ is the force on atom i imposed by the external environment, $u_2(r_i, r_j)$ is the intermolecular force between the two atoms i and j , u_3 is the three-body force between i , j , and k , and so on up to u_N which considers the entire system of N atoms interacting together. This approach obviously becomes unwieldy for a large system, so computer simulations must use less complex approximations (6:51).

The advantage of the pair potential method is that it is simple and straightforward (13:7). This method assumes that all forces beyond the second order terms are negligible. The overall force on each atom is, therefore, computed as a superposition of the forces on each two atom pair (6). The computer uses a truncation of equation (2.1), where, with no external forces:

$$U(1, N) = \sum_{\substack{i,j \\ i < j}} u_2(r_i, r_j) \quad (2.2)$$

The pair potential method necessarily assumes spherical symmetry, and has proven successful in simulations of noble gasses (13:7). Where electrons are less tightly bound, such as in metals, however, the pair potential method does not give satisfactory results and the force calculations must become more complicated.

The Embedded-Atom Method (EAM)

Because electrons in metals are less tightly bound than electrons in the noble gasses, the atomic environment can be modeled as nuclear cores surrounded by both their own electrons and by unbound electrons acting like an electron gas. The basic approach of the embedded-atom method is to calculate the force required to embed each atom in a host lattice of all the other atoms in the crystal (2:6443). Local electron density is computed from the electron density of the atom and the background density of the electron gas in the lattice (8:7984). Pair potentials are used to determine the repulsion between atomic cores, and the total force on each atom is the sum of the core-core repulsion and the force imposed by the unbound electrons (4:697).

The general embedded-atom method is not significantly more complicated to use than pair potentials, and the actual calculations are not especially more computer intensive. In addition, the embedded-atom method is successful at describing

more complex situations and is especially useful for studies of alloys as well as for pure metals (8:7984).

Daw and Baskes first suggested the embedded-atom method (2), and it has been successfully applied to hydrogen embrittlement in metals (3), phonons in transition metals (4), liquid transition metals (7), and to defects in metals (2). The success of the embedded-atom method is, in the final analysis, what justifies its use.

The first task of an embedded-atom algorithm is to determine the electron density at the location of each atom. This density is approximated by a superposition of atomic densities:

$$\rho_{n,i} = \sum_{j(\neq i)} \rho_j^a(R_{ij}) \quad (2.3)$$

where $\rho_{n,i}$ is the host electron density at atom i imposed by the other atoms in the lattice, $\rho_j^a(R)$ is the electron density contributed by atom j , and R_{ij} is the distance between atoms i and j (7:3410).

A known electron density can be applied to an embedding function $Fe_i(\rho)$ which gives the energy needed to embed atom i into the background density. A second function, $\phi_{ij}(R_{ij})$, is simply a pair potential which gives the core-core pair repulsion between atoms i and j .

The total system energy is then given by:

$$E_{tot} = \sum Fe_i(\rho_{n,i}) + \frac{1}{2} \sum_i \sum_{j(\neq i)} \phi_{ij}(R_{ij}) \quad (2.4)$$

(7:3409)

Values for the embedding function Φ and the core-core repulsion function ϕ have been obtained empirically by fitting physical properties. Daw and Baskes determined Φ and ϕ by using lattice constants, elastic constants, vacancy-formation energies, and sublimation energies in fcc and bcc phases (7:3410). In the computer, the values are stored in a table and splines are used to interpolate the specific values needed for the calculations. The file which contained the values for the copper interaction functions was provided with the original computer code.

Integration and Computation

Because each atom exerts force on its neighboring atoms, Newton's Second Equation of Motion:

$$ma = F \quad (2.5)$$

is rewritten:

$$m_i \frac{d}{dt} (v_i) = F_i = - \frac{d}{dr_{ij}} [u(i)] \quad (2.6)$$

where F_i is the force on atom i , a is acceleration, m is the mass of each atom, r_{ij} is the distance between atoms i and j , and $u(i)$ is the interatomic potential on atom i . Forces caused by the interatomic potential determine each atom's acceleration and displacement, the equations of motion are solved over time by using a fifth-order predictor-corrector algorithm. The forces are then summed and observable parameters are calculated.

Calculation of Parameters

All state functions, such as temperature and potential energy, are given by algorithms well established in molecular dynamics (10:10;13:28,29;17:10).

The stress tensor is of particular importance because it determines internal pressure, and that internal pressure influences atomic motions in subsequent time steps. The stress tensor, which describes the stress within the crystal, is determined by the motions of each atom multiplied by its weight and reduced by the interatomic attractions between the atoms. The result is a three-by-three stress tensor which is symmetric to the diagonal (1). Readers especially interested in the stress calculation should refer directly to the Baskes and Foiles computer program.

The potential energy of each atom is determined during the force calculations. System potential energy is a summation of the potential energy of each atom:

$$U = \sum_{i=1} u(i) \quad (2.7)$$

Kinetic energy is calculated by a straightforward:

$$K.E. = \frac{1}{2} m \sum_{i=1}^N v^2 \quad (2.8)$$

summed over each atom's calculated velocity.

Temperature is then calculated by:

$$\text{Temperature} = \frac{\text{const} \times K.E.}{\text{number of atoms}} \quad (2.9)$$

where the constant converts from units of energy to degrees Kelvin.

The volume is the product of the periodic lengths of the crystal in each of the three spatial dimensions. The periodic lengths are either fixed, as in a constant volume situation, or they are calculated as an additional degree of freedom when the volume is allowed to be dynamic.

Pressure is the sum of the diagonal of the stress tensor divided by three times the volume:

$$\text{pressure} = \frac{\text{stress}(1, 1) + \text{stress}(2, 2) + \text{stress}(3, 3)}{3.0 \times \text{volume}} \quad (2.10)$$

which can also be written as:

$$\text{pressure} = \frac{N k T}{V} - \frac{1}{3 V} < \sum_{i < j}^N r_{ij} \frac{du(r_{ij})}{dr_{ij}} > \quad (2.11)$$

where N is the number of atoms, k is Boltzmann's constant, T is temperature, V is volume and r_{ij} is the distance between atoms i and j .

Total energy is the sum of the kinetic and potential energies:

$$E = U + \text{K.E.} \quad (2.12)$$

The time average of any parameter, P , is simply the sum of that parameter divided by the number of time steps, M , used to get the sum:

$$\langle P \rangle = \frac{1}{M} \sum_{m=1}^M p(t_m) \quad (2.13)$$

Determining the Time Steps and Length of Runs

The optimum time step for each integration is a compromise between short steps for accuracy and long steps which save computer time and still generate meaningful data. A time step of 0.001×10^{-12} second (0.001 picosecond or 10^{-15} sec) gives the required compromise (17:16).

During each run to generate data on the reactions of a copper crystal, the results from the first 300 time steps (0.3 ps) are not added into the final averages in order to let the crystal react to the changes in stress imposed by the experiment. This delay ensures that the entire crystal is equilibrated with a Maxwell-Boltzmann distribution of velocities based on temperature. The crystal is then left undisturbed for two or five ps, depending on the nature of the particular computer run. Both instantaneous and average values of important parameters, such as temperature and pressure, are printed at intervals selected by the operator in order to show trends and to establish that there are no discontinuous jumps in the data which would indicate inaccuracies in calculations or the use of excessively long time steps. Final averages give an indication of the state of the copper over the entire undisturbed period.

Alternative Configurations

The Baskes and Foiles code allowed further operator input in order to specify the type of problem to be solved. Different parameters such as volume or total energy can be held constant depending on the atomic ensemble to be generated by the experiment. One choice is between fixed or dynamic crystal boundaries. In order to use changes in volume as an indication of the state of the copper crystal, the researcher selected dynamic boundaries for all computer runs.

Another choice was between a constant energy problem and a temperature fixed problem. In the constant energy situation, the internal conditions of the crystal reflect the reactions of the crystal as if no energy is allowed to enter or leave the simulation. In the temperature fixed situation an external temperature is applied to the crystal and energy can flow into or out of the crystal and into the surrounding environment. All computer runs to gather data on the state of copper were done with the temperature fixed so that the crystal's internal energy could be affected and so that the state could change.

Time Saving Steps

In order to save computer time and still obtain useful answers, time saving steps are available. Rather than compute the effect of every atom on every other atom, for example, the computer can develop a neighbor list for each atom. Neighbors are those atoms located within a specified cutoff distance, also called a neighbor interaction sphere as in Figure 2-4.

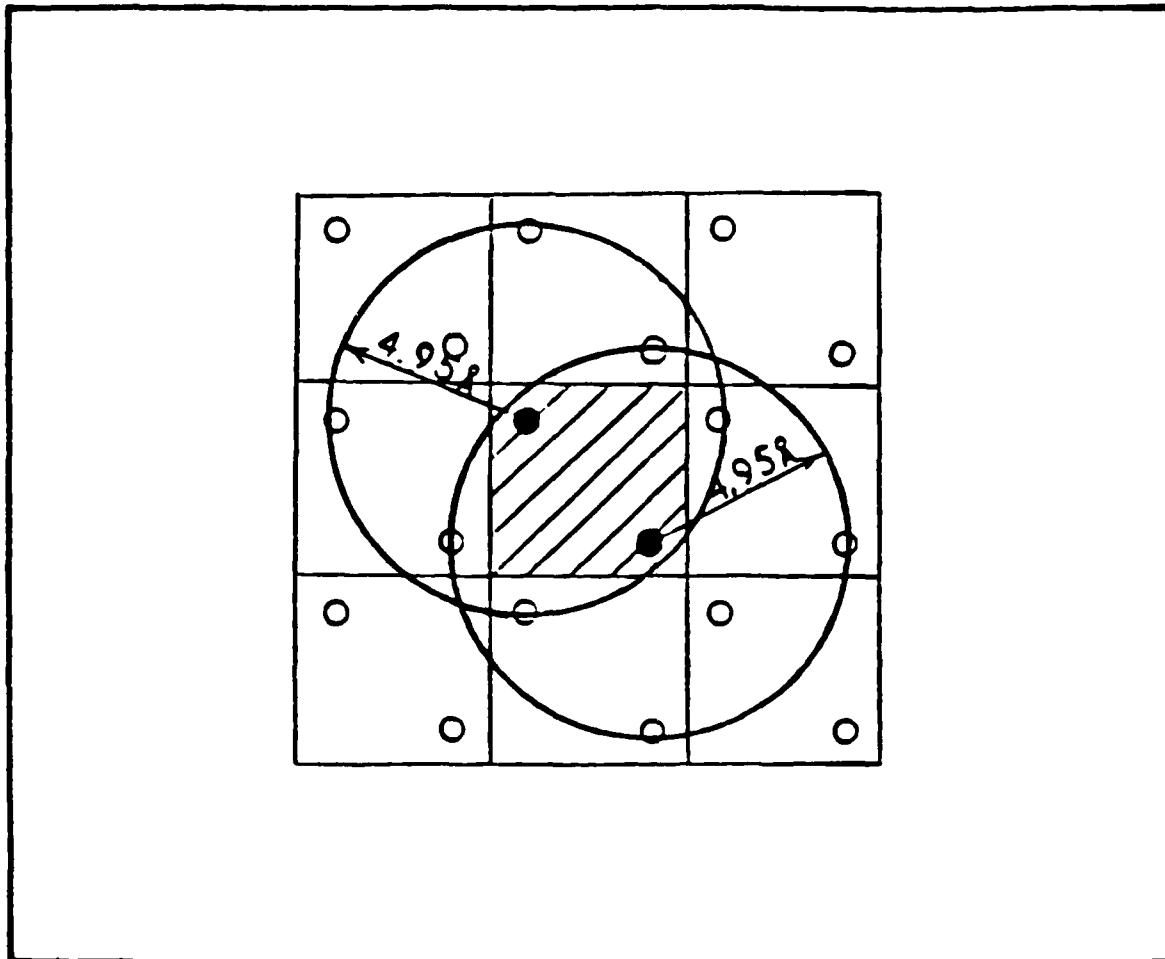


Figure 2-4. Neighbor Interaction Spheres (in 2 Dimensions)
(from 17:77)

Atoms outside the cutoff are assumed to exert negligible force on the individual atom under evaluation; hence, by definition within the computer, atoms outside the cutoff radius do not directly interact.

Either the original neighbor list can be used through the rest of the computer run or the list can be revised when any atom is displaced more than a threshold distance. Both the cutoff distance and the threshold distance can be specified by the operator or can be calculated by the computer.

In order to avoid repeating calculations, steps can be eliminated by applying Newton's Third Law of Motion: Every action has an equal and opposite reaction (11:66). This means that every atom affects every other atom with the same force in the opposite direction and eliminates the need to calculate the forces between atoms j and i if the forces between atoms i and j are already known.

An advanced time saving step is to use a different computational approach, a parallel processing computer. This approach is discussed in the next chapter.

III. PARALLEL PROCESSING

In order to expedite program execution, the molecular dynamics program was modified to run on a parallel processing computer. This chapter discusses the definition of parallel processing, how certain molecular dynamics calculations are well suited for solution by parallel processing techniques, and how the specific embedded atom program was modified to run in parallel.

Parallel Processing Computers

The conventional approach within a computer is serial processing, the practice of a single computer working through a problem one step at a time. Parallel processing is often mentioned as one of the next great advances in data processing because may allow many steps to be done at once (5:13). The basic idea is to have a controlling computer, also called a host, manage several computers, called nodes, that can communicate with the controller and with each other. The node computers can be given the necessary information, complete their given tasks, and send the results to other nodes or to the host. With this arrangement, each node can work part of a larger problem and many nodes working at once can reduce the time needed to reach a solution.

Modification of any program from a serial to a parallel approach requires the programmer to select the processes that can be solved in parallel and the communications arrangements

between the host and the nodes. Processes such as solving an array, for example, can often be split so each node solves part of the array and communicates its results. Many host/node communication arrangements are possible, ranging from rings, where each node communicates with two adjacent nodes, to trees, a hierarchical arrangement, to multidimensional cubes, also called hypercubes (9:73). Figure 3-1 illustrates some of the possible communications paths.

Molecular Dynamics as a Parallel Process

As indicated in Chapter Two, the force and neighbor list calculations are the most computer intensive parts of a molecular dynamics program. Fortunately, both types of calculations are suitable for parallel computing by assigning each node an appropriate share of the atoms in the experiment.

Before the force on any given atom can be calculated, for example, the neighbor list for that atom must be developed by comparing that atom's position with the positions of every other atom within a cutoff distance as shown in Figure 3-2. As a result, the neighbor lists within a serial computer can grow to be both large and complicated as the assigned number of each neighboring atom can either be stored in an individual array for each atom or it can be stored in one big array with each atom assigned an index number which tells where its neighbors are listed. Parallel processing can speed this by finding the neighbor lists for many atoms once and can, in turn, simplify the computer memory process as each node will

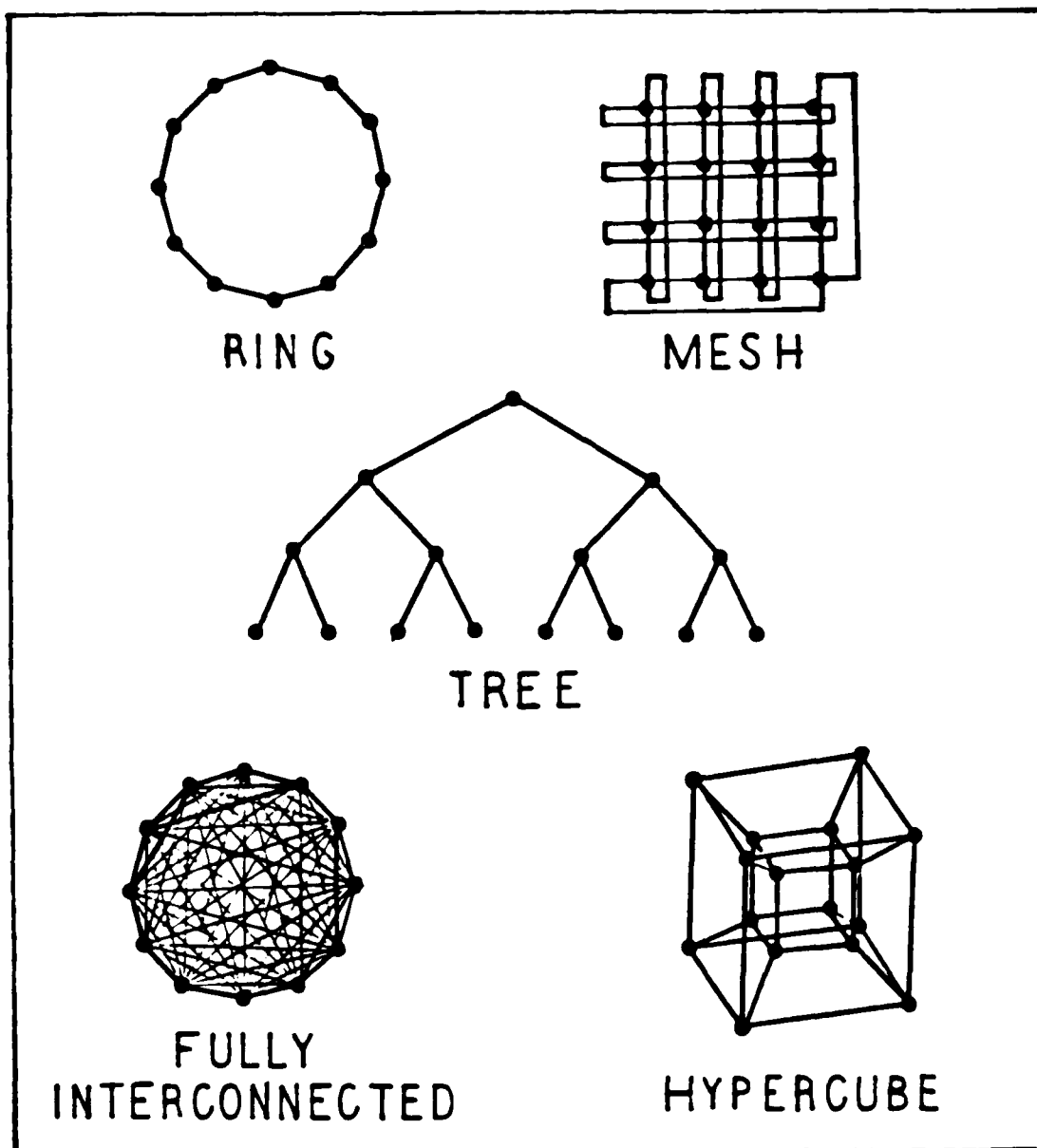


Figure 3-1. Possible Node Communication Paths
(from 9:73)

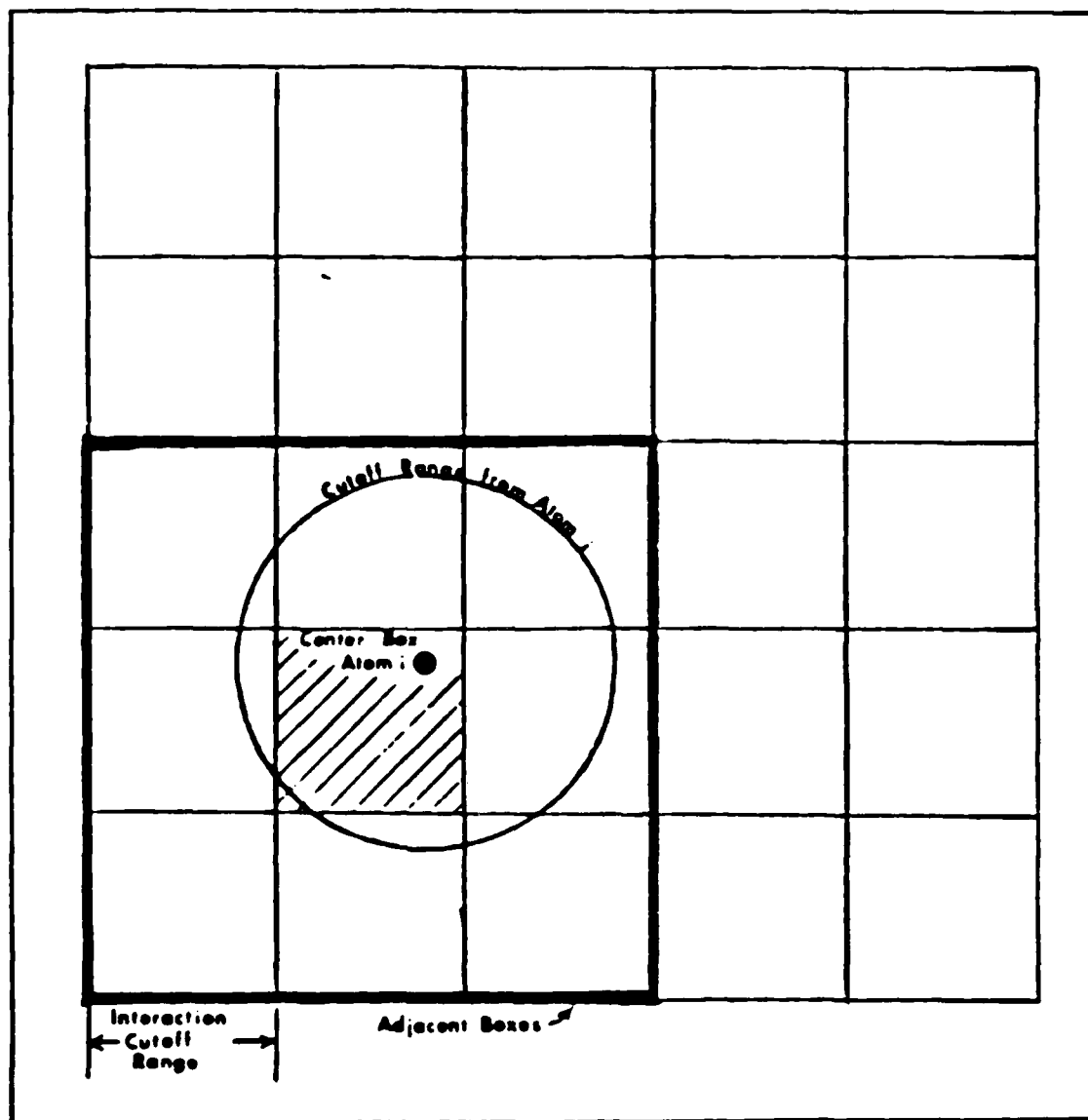


Figure 3-2. The Interaction Cutoff Range and Adjacent Lattice Boxes
(from 17:30)

need to store either fewer individual neighbor arrays or one much shorter neighbor array. For example, the node which works atoms eight through fifteen stores the neighbor lists for only those atoms. That node does not need to waste time or memory by calculating and listing the neighbors of atoms outside its assigned area.

Force calculations can be made much faster by the use of a similar approach. Each node can calculate forces on an appropriate share of the atoms in the lattice and report its results back to the host. The host combines those results in order to continue the calculations for the particular time step.

The Embedded-Atom Solution in Parallel

An embedded-atom program modification which uses parallel processing and which gives complete solutions follows the basic approach outlined above. The communications plan is to have the host send identical information to all nodes. Each node first assigns itself several atoms within the copper crystal. Each individual node finds the neighbors of its assigned atoms, then determines the local electron density for each assigned atomic position. The electron density is used to find the embedding energy of each atom, and finally each node calculates the core repulsion between its assigned atoms and all other atoms on its neighbor list. The results are passed back to the host and the nodes are ready to begin again.

The obvious advantage of the parallel approach is that the work is divided among several processors and the forces and neighbors for many atoms are solved at once. This process is illustrated by Figure 3-3. Adding nodes does not divide the total computer time by the number of nodes, however. Communications between host and nodes becomes time consuming as only one node can report back to the host at one time and as the host computer must reinitialize the arrays used for communications after the reports are received. The potential advantages of parallel processing become a trade off between time saved by simultaneous operations and time lost to increased communications requirements.

Unfortunately, also, the parallel processing computer was not continuously available during this research period and the program modification process proved to be quite time consuming. Because the iPSC cannot run in a background mode, parallel computer runs were restricted to provide validation and timing information rather than provide scientific data on the copper crystal. Timing advantages are discussed in Chapter Four -- Experiments and Results. The scientific data used in the investigation of the equation of state was generated on the ELIXI system with its sequential processor.

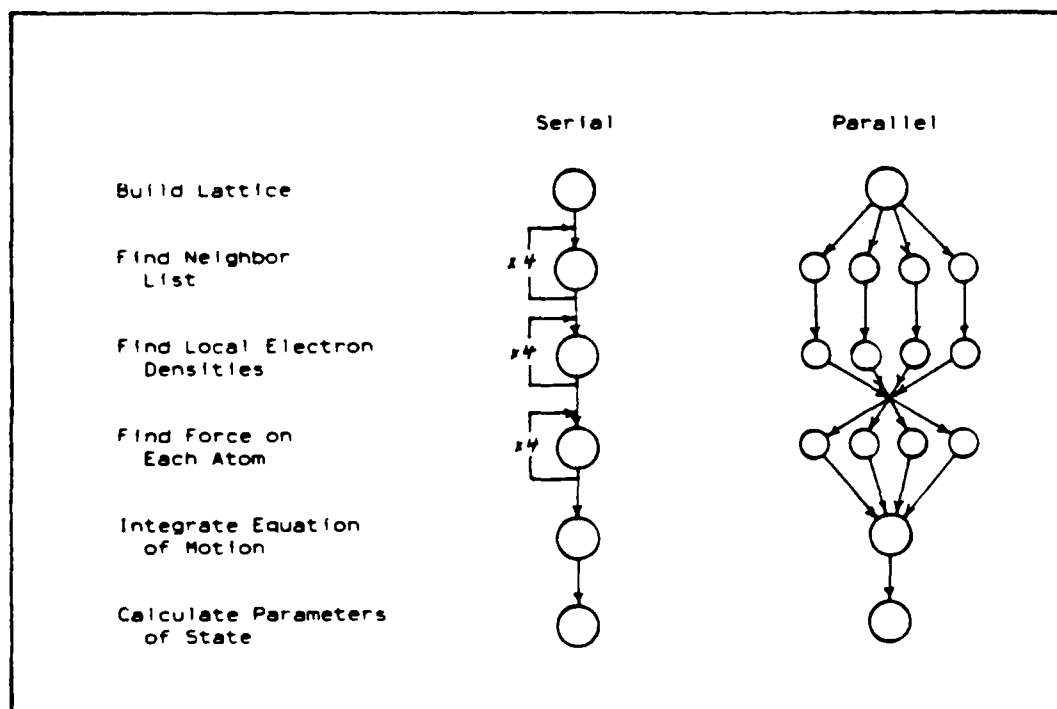


Figure 3-3. A Parallel Processing Block Diagram

IV. EXPERIMENTS AND RESULTS

The aspect of copper studied by the modified molecular dynamics programs was, in general, the equation of state, and specifically, the transitions from solid to liquid and back to solid. This chapter will review the equation of state for copper, the methods employed, the different approaches used to study aspects of the transition problem, and the results of those studies. The chapter concludes with a discussion of the results of the parallel processing experiments.

The Equation of State for Copper

The equation of state:

$$\bar{p} V = N k T \quad (4.1)$$

applies to an ideal gas, where \bar{p} is the mean pressure, V is the volume of the system in question, N is the number of molecules in the system, k is Boltzmann's constant, and T is the temperature of the system (14:125). A second equation gives E , the total system energy:

$$E = \frac{3}{2} N k T \quad (4.2)$$

Equations of state are important because they relate parameters that can be measured in the laboratory. Given any three parameters in this case, the remaining two can be readily calculated.

Copper, however, is not an ideal gas and the equation of state is more complex. Rather than derive the equation, this

study investigates the phase transitions of copper as external pressure was held constant and a selected external temperature was applied and held constant through each entire computer run. The resultant changes in internal pressure, volume, internal temperature, and potential energy were observed. A change from the solid to a liquid state was observed when the volume, linear displacement, and potential energy, as plotted against the independent variable of internal temperature, made a discontinuous jump from what was otherwise a normal pattern of expansion and contraction driven by temperature.

A risk of using discontinuities in volume and energy is that the transition may not be readily apparent if the crystal is caught in a partially melted/frozen state where the average values fall between the values for liquid or solid. Although those points are hard to analyze at first, such ambiguous data points provide opportunities for further study because the final state of the atoms of that run can be used to begin another run for a longer time or at a slightly different temperature. The combined information from such multiple runs can provide increasingly specific information on melt and freeze temperatures.

Another problem of the molecular dynamics approach is that artificial superheating and supercooling generally cause one to miss the actual melt or freeze points. Several aspects of molecular dynamics combine to make this a possibility that must be recognized and not allowed to invalidate otherwise good results from computer studies. Because the crystals

simulated in the computer are necessarily small, time steps are too small to be directly observed, and the mathematics allows pure crystals in perfectly defined conditions, some results not readily observable in the laboratory must be accounted for. Superheating was a problem in the first attempt to melt the copper, and supercooling occurred consistently in the cooling/freezing experiments.

The entire series of computer experiments on the ELIXI serial processing computer used a copper crystal of 256 atoms. The initial run began at 293° Kelvin (K), room temperature, and with a lattice constant of 3.165Å, in agreement with laboratory data (19:12). Next, the copper crystal was heated in order to find its melting point. After the copper melted, the crystal was cooled until it froze. The frozen crystal was then reheated to see if the melt point changed, and finally the remelted copper was again cooled in an attempt to repeat the freezing. All runs applied an external temperature which varied from run to run but was held steady throughout each run, and a constant external pressure of zero bars. The internal temperature, energy, and volume of the crystal was recorded and averaged over several thousand time steps.

Validation -- Copper at Room Temperature

The first computer run used the structure of an fcc crystal and the lattice constant of copper to build the matrices that describe the positions and velocities of every atom of a copper crystal. Each atom was assigned random motions with a

Boltzmann distribution based on 293° Kelvin, the external temperature was held at 293° K, and the system was given a standard 0.30 ps (300 time steps) to allow the system to equilibrate. The crystal was given flexible boundaries and the observable parameters were averaged over two thousand time steps (2.0 ps). The average volume was used to compute the average lattice constant and that distance was compared to the lattice constant of copper under laboratory conditions.

The average volume found by the computer was 3067.86 Å³ (± 0.079 Å³) which gives an average lattice constant of:

$$\frac{(3067.86)^{1/3}}{4} = 3.633 \text{ Å } (\pm 0.00003 \text{ Å})$$

which is within 0.50% of the accepted lattice constant of 3.615 Å at room temperature (20:12).

With the computer results for conditions at room temperature in agreement with actual laboratory data, the volume of 3067.86 Å³ at 293° was used as the baseline to compare volumes at other temperatures. Three dependent parameters -- volume, linear expansion, and potential energy -- are used in the following sections. In order to show the change in the state of the copper crystal, the three dependent parameters are plotted against average internal temperature for each computer run. Discontinuous changes in the three parameters indicate a change of state and all three plots must agree for a change of state to be unambiguous.

Determination of the Equation of State

The First Series of Heating Runs

The first task was to determine if the copper would melt at the accepted temperature of 1357.6°K (18:77). This was done by first building a computer model of a crystal at 293°K , heating it to 300°K and holding it there for 2.3 ps while using the internal parameters of the last 2.0 ps to calculate overall average values. The crystal was then heated to 600°K , then 900°K , 1200°K , 1500°K , and finally 1800°K in separate computer runs before it melted. This melting point is higher than found in the laboratory and it originally put the entire approach in doubt.

A Second Form of Validation

A second means of validation was then applied by comparing the percentage of linear expansion calculated in the simulation against accepted tabulations (18:77). The results are plotted on Figure 4-1. The original data points at 1200°K and 1500°K were higher than the prediction, but after the crystal was held at each temperature for 5.3 ps, the linear distance lowered to within excellent agreement with the expansion predictions. These points are also plotted on Figure 4-1 and they help confirm that the computer gives reliable information.

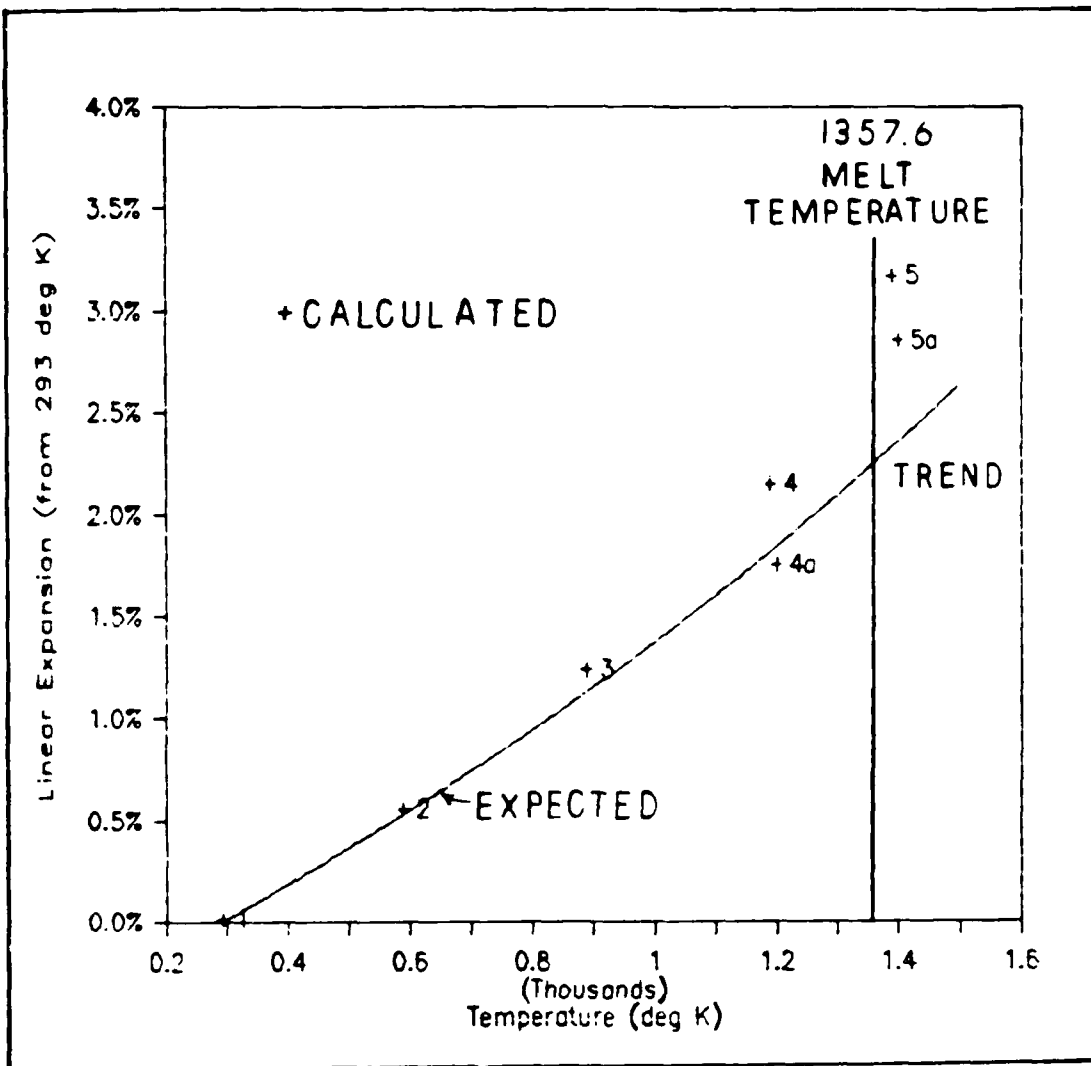


Figure 4-1. Computer Calculated and Expected Linear Expansion vs. Temperature
(Expected values from 18:77)

Melting After Superheating

The high melting temperature of the first series of runs is, therefore, an apparent example of superheating. The results are shown on figures 4-2, 4-3, and 4-4, and they give unambiguous agreement that the change of state occurred between 1500° K and 1800° K. All three graphs include data points at 1950° K and 2100° K to establish trend lines for the liquid state.

In an attempt to break the superheating and melt the crystal, the copper was held at 1500° for 5.3 ps, but no change of state was observed. Heating the crystal to 1550° and 1600° also did not cause a change of state. Temperature changes from 1500° to 1650° and from 1600° to 1650° K melted the copper as shown on Figures 4-5, 4-6, and 4-7.

To summarize the first series of heating the copper crystal, the simulations resulted in superheating the crystal past the accepted laboratory melting point until it melted between 1600° K and 1650° K.

Freezing the Melted Crystal

Obviously an actual liquid would not hold its shape; a melted copper crystal would form a molten puddle and its atoms would lose the order imposed by the crystal lattice. The computer simulation, however, allows the lattice to maintain its order even though the crystal has expanded and the increased distances between atoms allows the interatomic forces to decline enough to allow the change from solid to

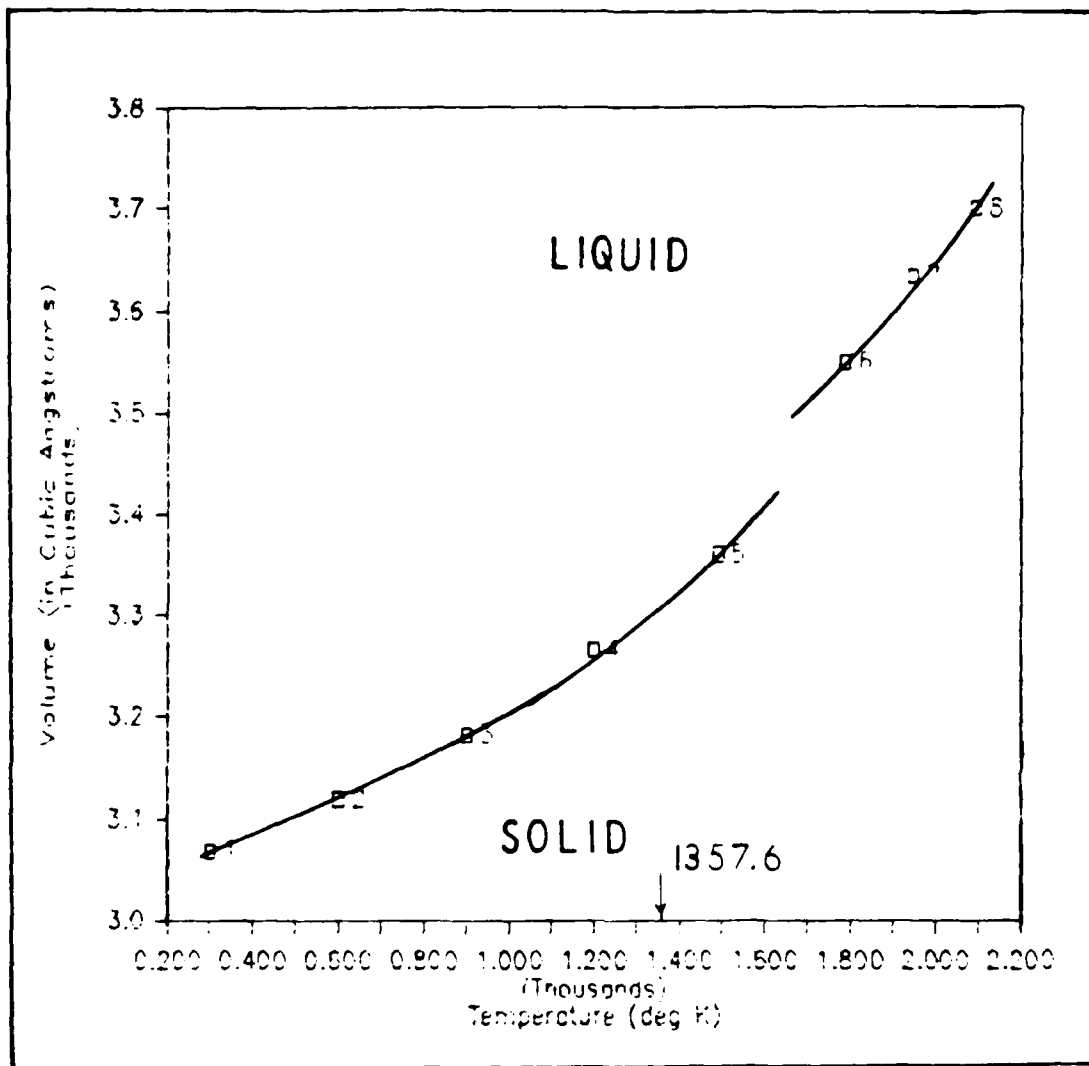


Figure 4-2. Volume of the Copper Crystal as it is Melting

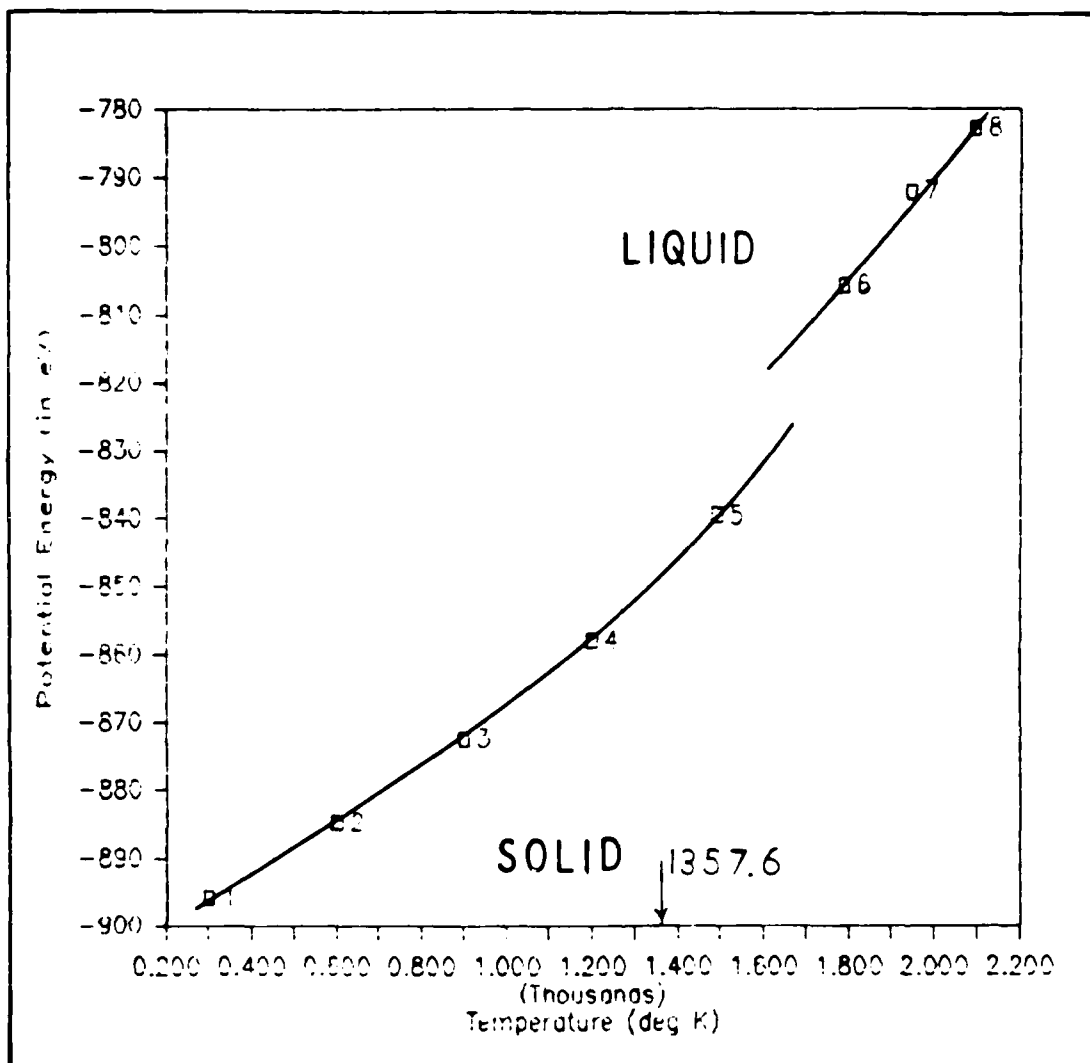


Figure 4-3. Potential Energy of the Copper Crystal as it is Melting

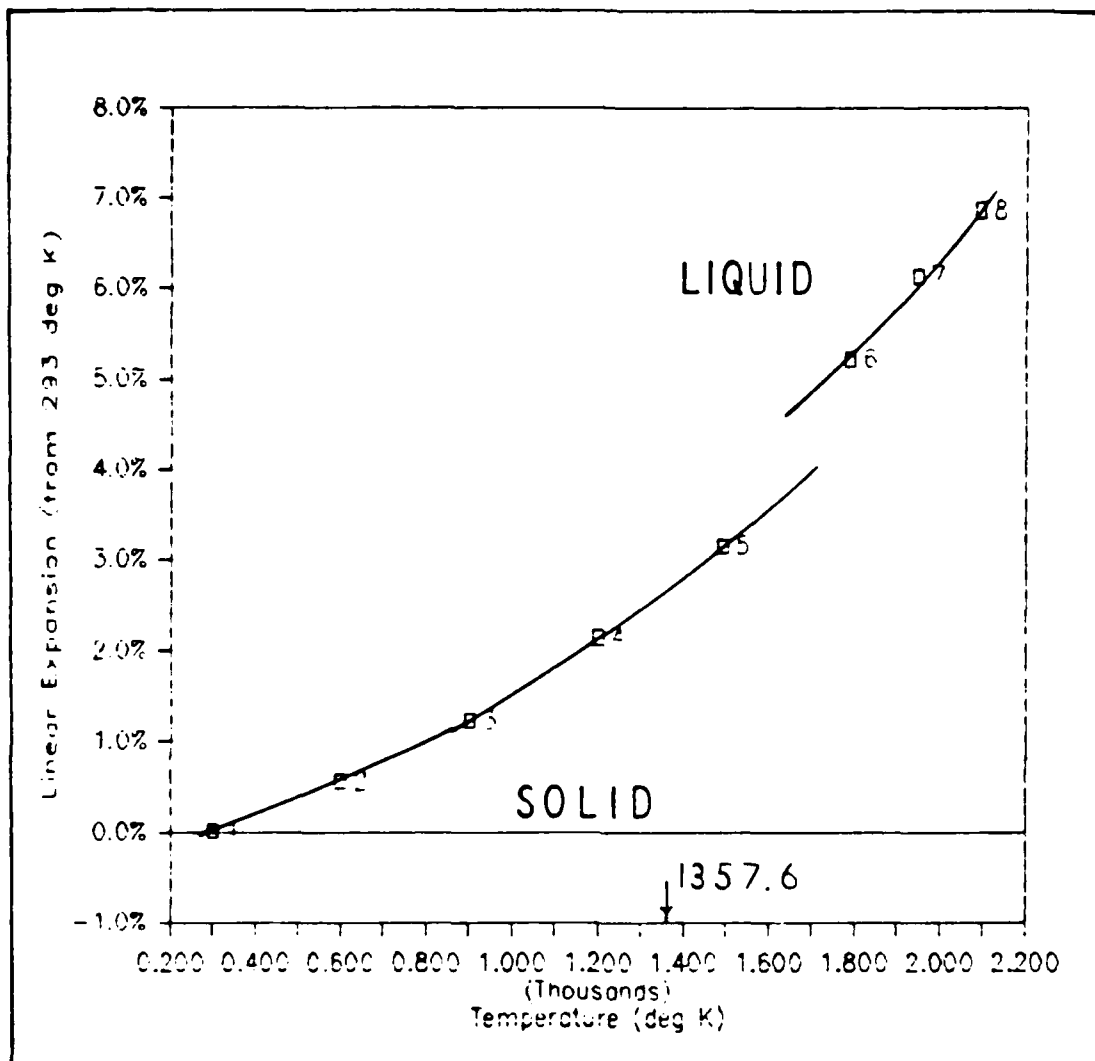


Figure 4-4. Linear Expansion of the Copper Crystal as it is Melting

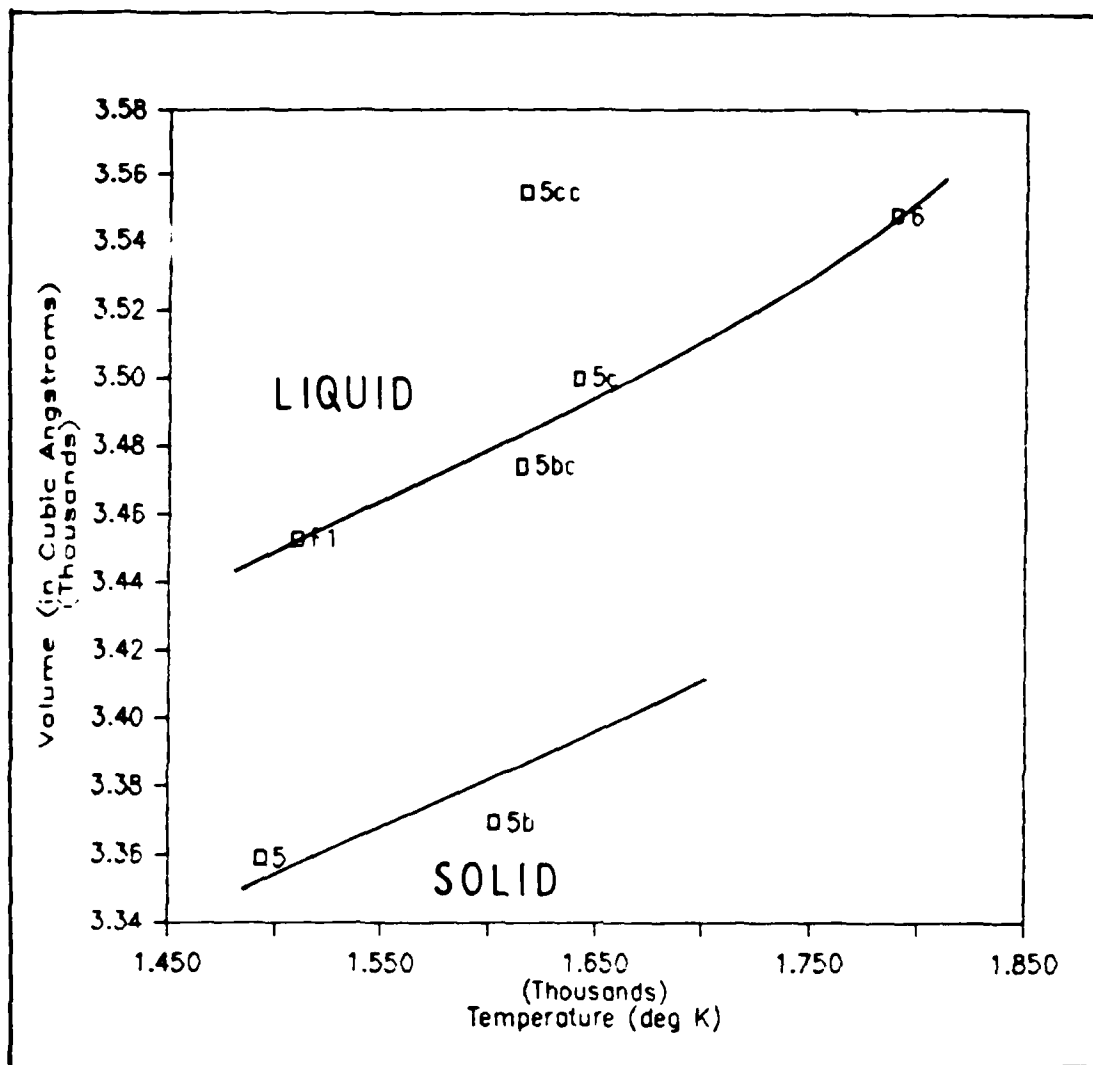


Figure 4-5. Volume of the Copper Crystal at the Melt Temperature

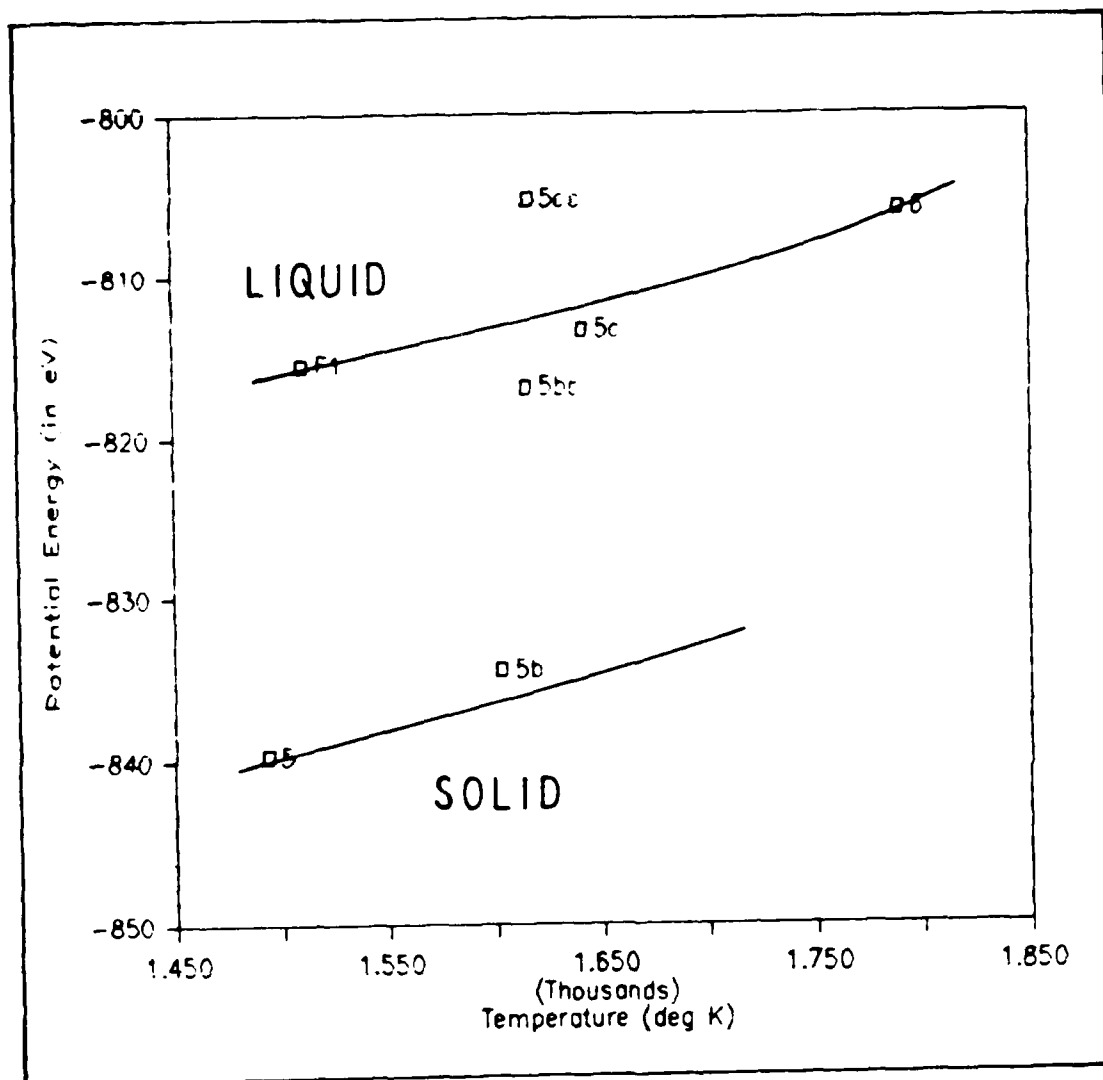


Figure 4-6. Potential Energy of the Copper Crystal at the Melt Temperature

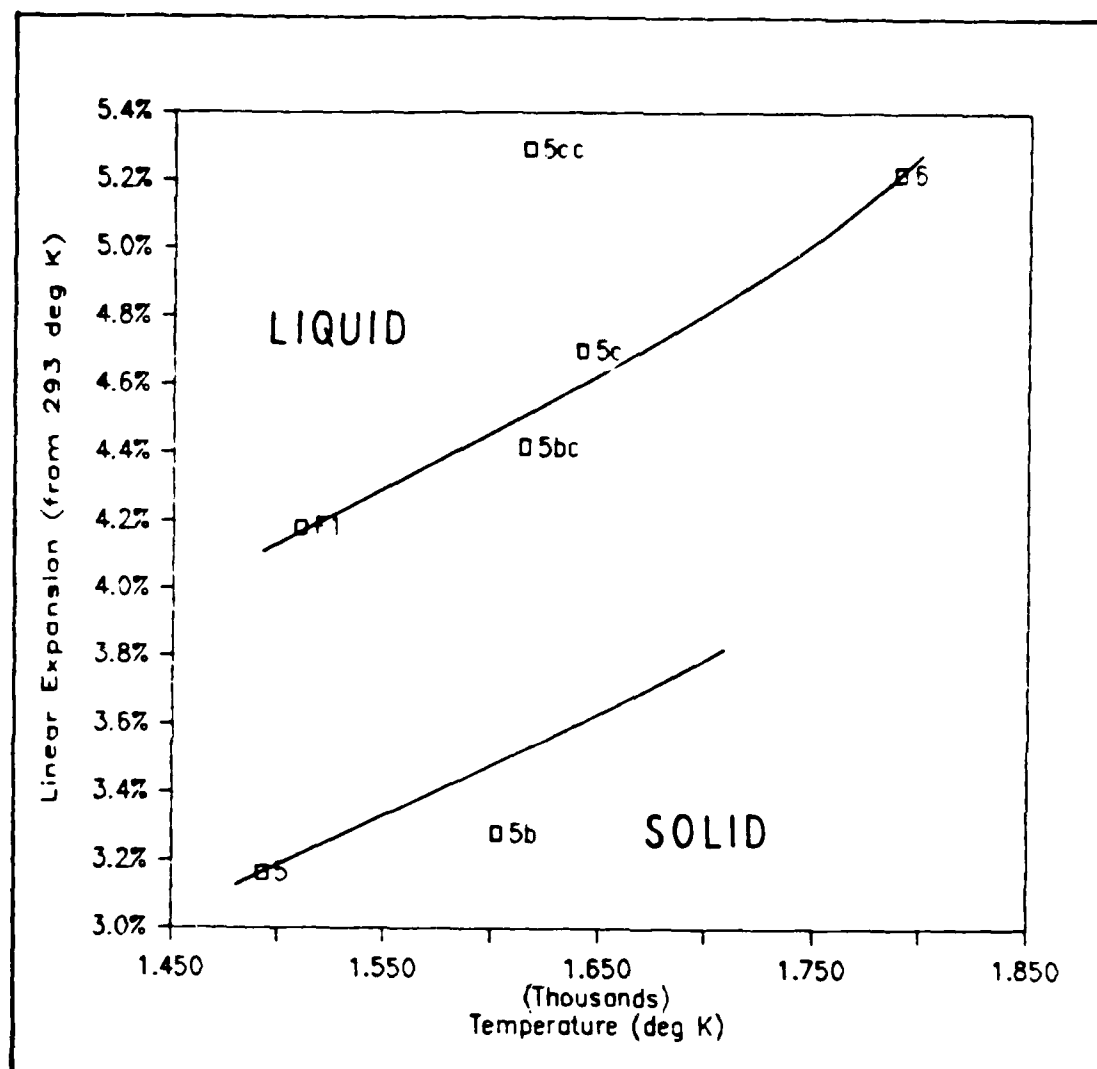


Figure 4-7. Linear Expansion of the Copper Crystal at the Melt Temperature

liquid. Hence, the atoms in the simulation retain their same order and crystalline structure when the temperature drops enough to let the crystal freeze.

Starting with the molten crystal at 1800° K, the external temperature was reduced to 1500° K, then 1200° K, 900° K, and 600° K with no observed change in state. The metal would not readily freeze, and this is an apparent case of supercooling. A computer run of 5300 steps (5.3 ps) at 1200° K gave virtually the same results as the original 2300 step run. Holding the temperature steady over a total of four long computer runs at 900° K, however, let the crystal freeze. The time involved in the freezing process was a total of 21,200 time steps or 21.2 ps. The changes are shown in figures 4-8, 4-9, and 4-10.

Once the crystal started to freeze, heating it while it was freezing managed to speed the freezing process. Because the atoms are moving relatively slowly at the relatively low temperature of 900° K, they will take longer to move back into the closer positions of a solid. The crystal in a partially frozen state at 900° K was heated to 1200° K, where it first remained in its partially frozen state, and then froze solid as it was held at 1200° K for one long (5.3 ps) computer run. At an equivalent partially frozen state at 900° K, the crystal took twice as long to freeze to a solid.

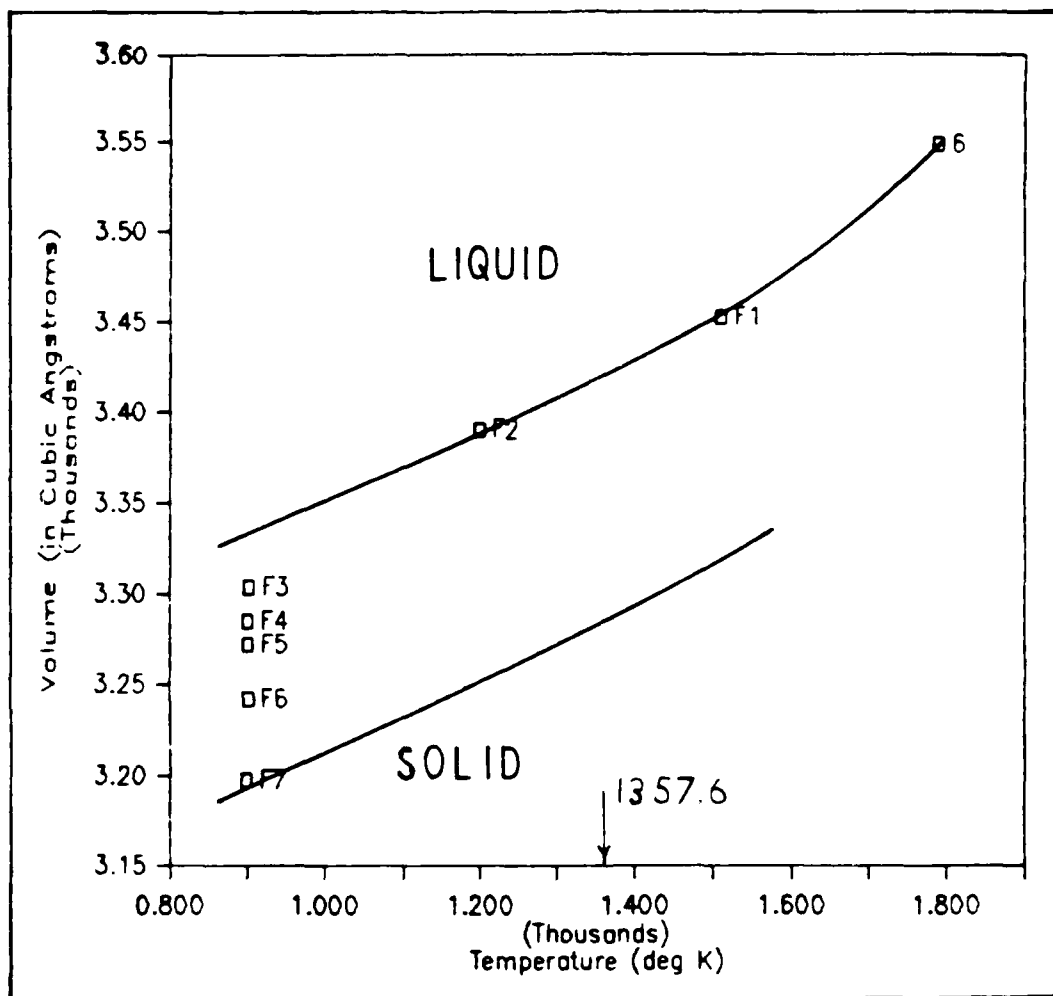


Figure 4-8. Volume of the Copper Crystal as it Freezes.

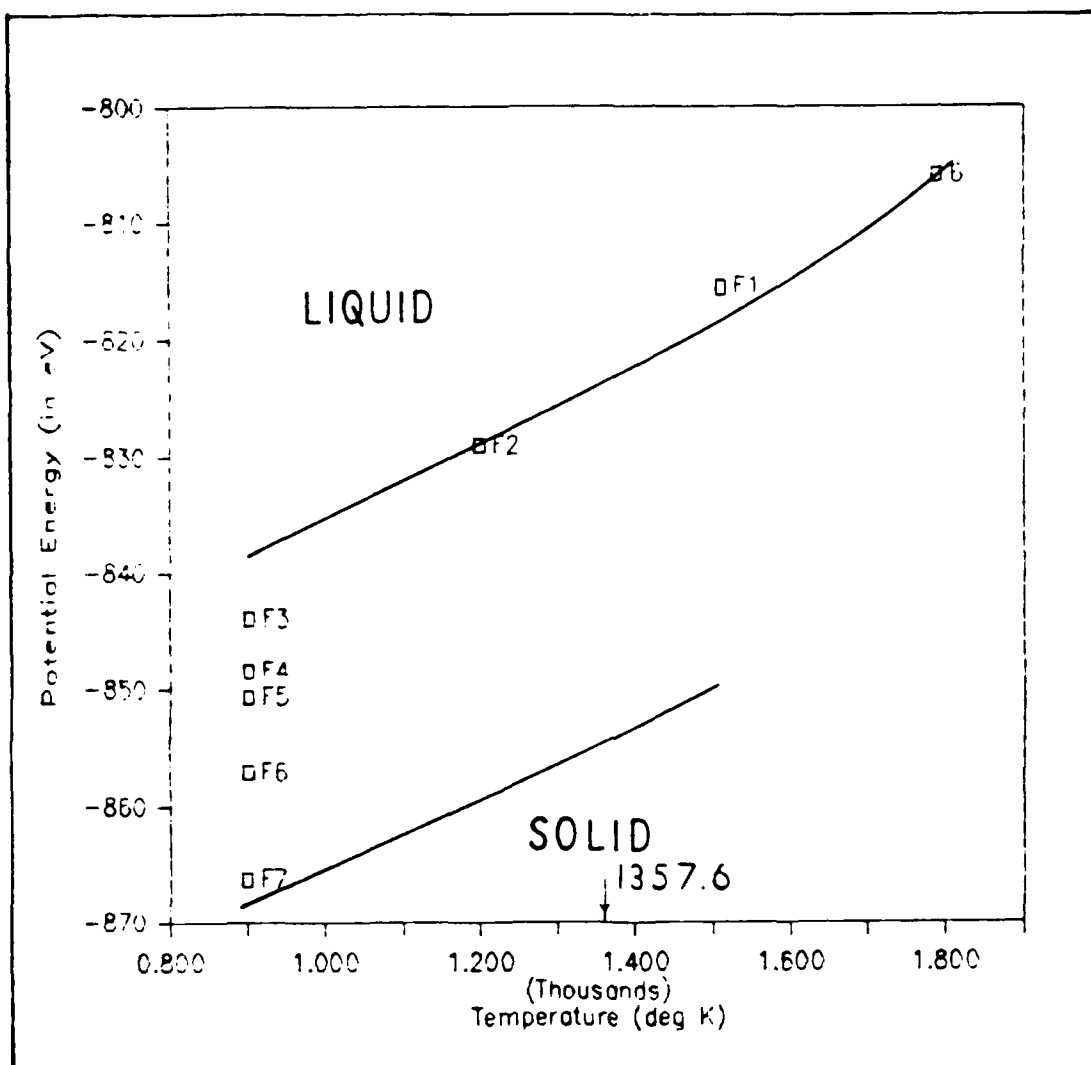


Figure 4-9. Potential Energy of the Copper Crystal as it Freezes

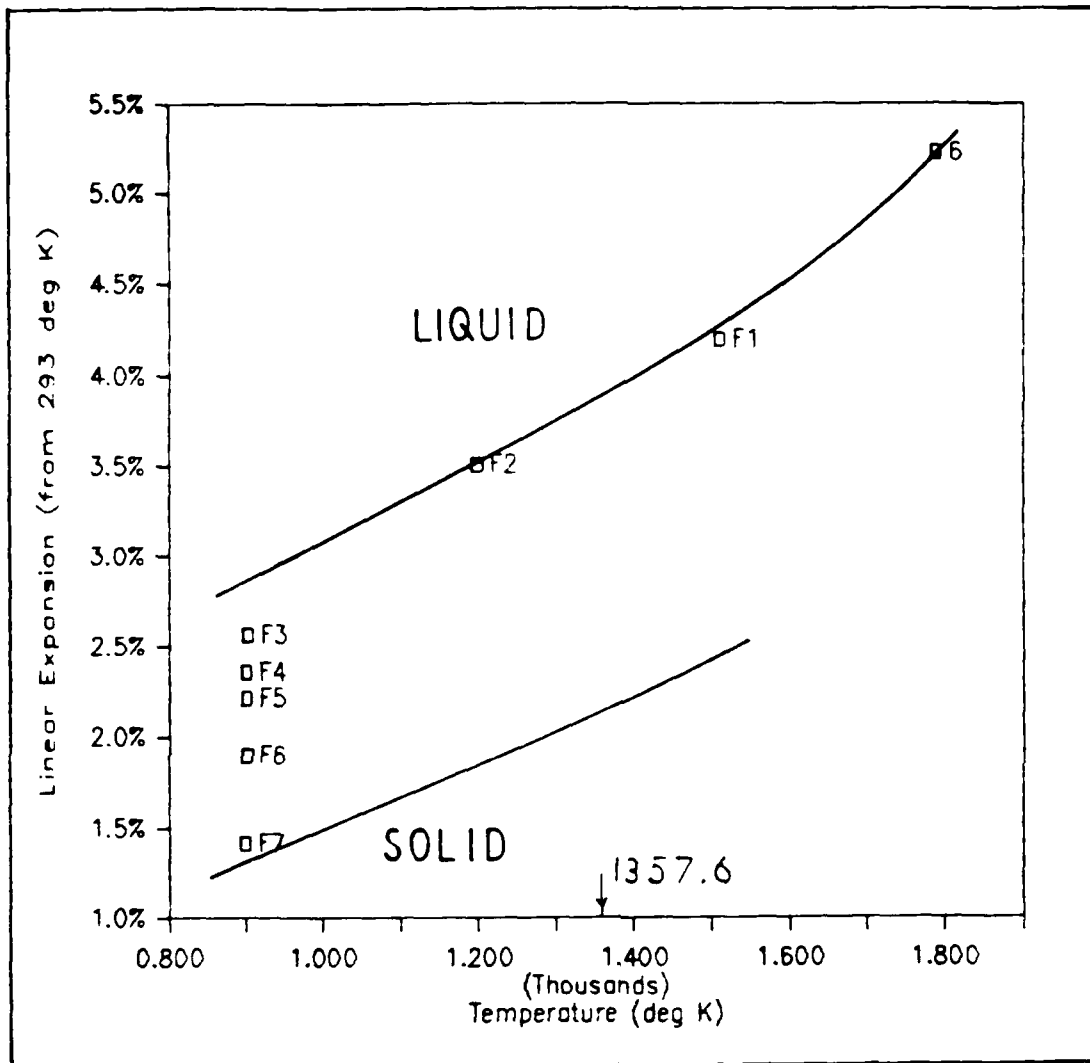


Figure 4-10. Linear Expansion of the Copper Crystal as it Freezes

Melting the Crystal Again

The refrozen crystal at 1200° K was again heated in an attempt to either repeat the superheating or find a more realistic melt temperature. The first run of this series, to 1500° K melted the crystal. The two temperatures of 1200° K and 1500° K bracket the known melt temperature of 1357.6° K.

The next run used the same start point at 1200° K and heated the crystal to 1400° K. The crystal melted again. Heating the 1200° K crystal to 1350° K gave a data point slightly above the trend line for the crystalline state, and holding that crystal at 1350° K, 7.6° K below the known melt point of copper, the internal temperature averaged 1347.78° K and the observable parameters fell between the trend lines for solid and liquid.

In order to confirm that the copper was transitioning near the accepted laboratory melt point, the slightly melted crystal at 1350° K was cooled to 1300° K, where it remained solid. These results bracket the melting point of copper between 1300° K and 1400° K and at least imply that the melt point is very near 1350° K. These results are shown in Figures 4-11, 4-12, and 4-13.

With the remelted copper at 1400° K, there was one last attempt to freeze it near the known freeze temperature. The crystal was cooled to 1300° K, but the data points remained along the liquid trend lines. The crystal had supercooled again.

Based on these results, the melting temperature of copper predicted by the simulation is 1350° K, plus or minus 50° K.

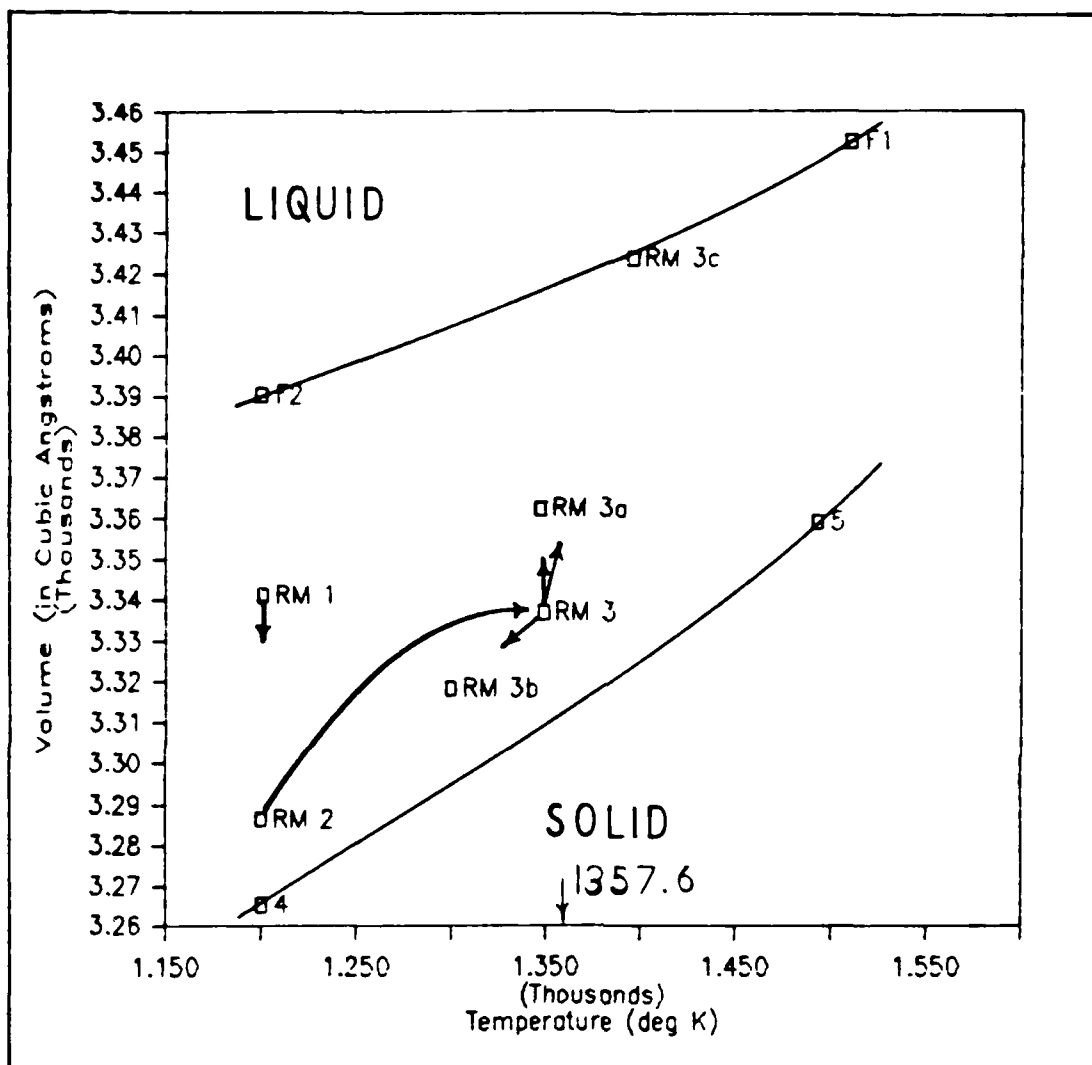
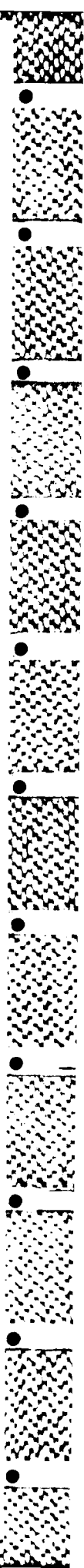


Figure 4-11. Volume of the Copper Crystal at the Actual Melt Point
The arrows represent how the crystal reacts to different temperatures.



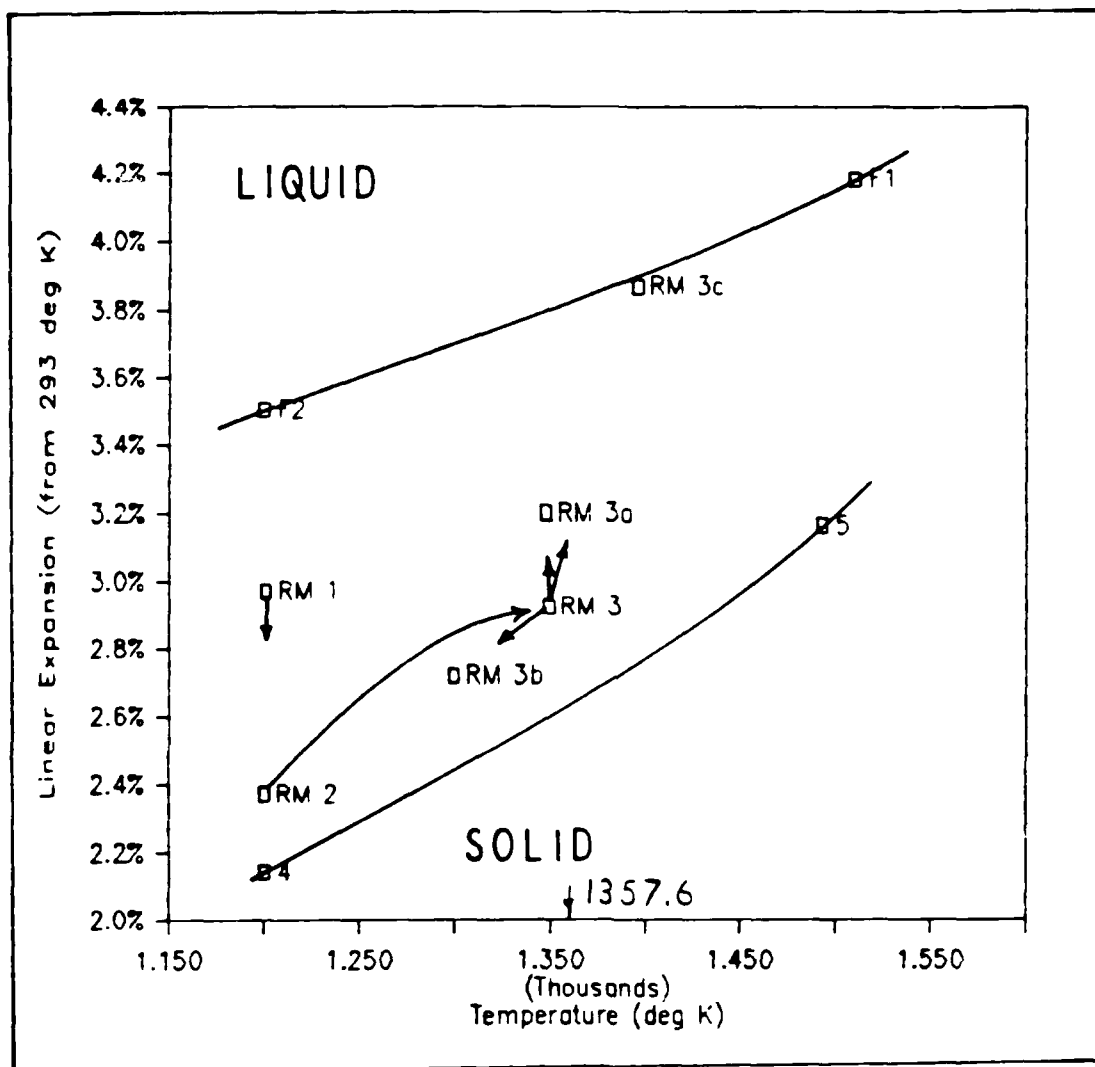


Figure 4-13. Linear Expansion of the Copper Crystal at the Actual Melt Point
The arrows represent how the crystal reacts to different temperatures.

Parallel Processing Results

Diagnostic devices available on the UNIX operating system show that from 96% to 98% of the central processing unit (cpu) time needed to solve a molecular dynamics problem on a conventional linear processing computer is spent doing the force and neighbor calculations. To speed the solution process, the force and neighbor subroutines were exported to the nodes of the iPSC parallel processing computer. Test programs were run using different numbers of nodes in order to show the proportionate time advantages of dividing the calculations among the nodes.

In order to keep the timing results between the serial and parallel computers comparable, both types of computers worked with identical 256 atom copper crystals held at a constant external temperature and with flexible boundaries. Programs which make comparable numbers and types of force and neighbor calculations were run on both types of computers.

The original computer code used special timing functions available on the Cray supercomputer. Computers based on the UNIX and EMBOS operating systems are not able to measure time with the same sophistication. Relatively simple functions were used to measure the time required to solve the entire atomistic simulation problem and to measure the time spent in the subroutines exported by the parallel computing process. A necessary assumption for this section is that the timing routines added to the original computer code gave sufficiently accurate timing information.

A simple comparison of total time spent on each program would not give meaningful data because the ELIXI and iPSC computers run at different speeds, and because total processing time depends on the level of multitasking which is beyond the control of the experimenter. Therefore, in order to establish potential time savings, the critical difference between serial and parallel computer runs was the proportion of time spent in the subroutines exported to the parallel computer nodes. The results and the calculations below establish that the parallel approach has the potential to save considerable computer time.

In order to make complete and accurate comparisons the serial UNIX programs were run in both a normal and an unfavorable configuration which did not use all available time saving steps. The normal configuration updated the neighbor list only as necessary. Cumulative time spent in the force and neighbor subroutines was 96% of the total time of the computer run. An unfavorable configuration was run where the neighbor list was updated at every time step. The force and neighbor subroutines used 98% of the total computer time.

After the program was rewritten for the parallel computer, experiments were run to obtain similar timing information. Unfortunately, the parallel processing modification at this time defaults into a mode which updates the neighbor list at every time step. Corrections to this problem have yet to work. Because neighbor list recalculation is a time consuming step, the only comparison fully substantiated by experiment at

this time is between the proportion of time used by the force and neighbor subroutines in the parallel computer runs and the 98% proportion used by the serial computer with the unfavorable program configuration.

The proportion of time spent in the subroutines exported to the nodes is plotted on Figure 4-14. This graph shows that the proportion of time spent doing force calculations (when the neighbor list is updated at every time step) decreased from an average of 98% for a serial computer to 85% for a parallel processing computer with thirty-two nodes. Because the force and neighbor subroutines take up most of the computing time, these savings promise considerable overall time savings.

When the serial computer did not recalculate the neighbor list at every time step, the force and neighbor subroutines took 96% of the total computer time, which gives the parallel processor a reduction of 11%. The calculations in Appendix A establish that the 11% time difference can reduce total processing time by a factor of 3.75 between processors of equal speed and tasking.

These results establish that the time required to reach a solution can decrease dramatically as the programming is improved and as parallel computer techniques become more sophisticated. The next chapter discusses how these results can be applied in order to make parallel processing a powerful addition to atomistic simulation techniques.

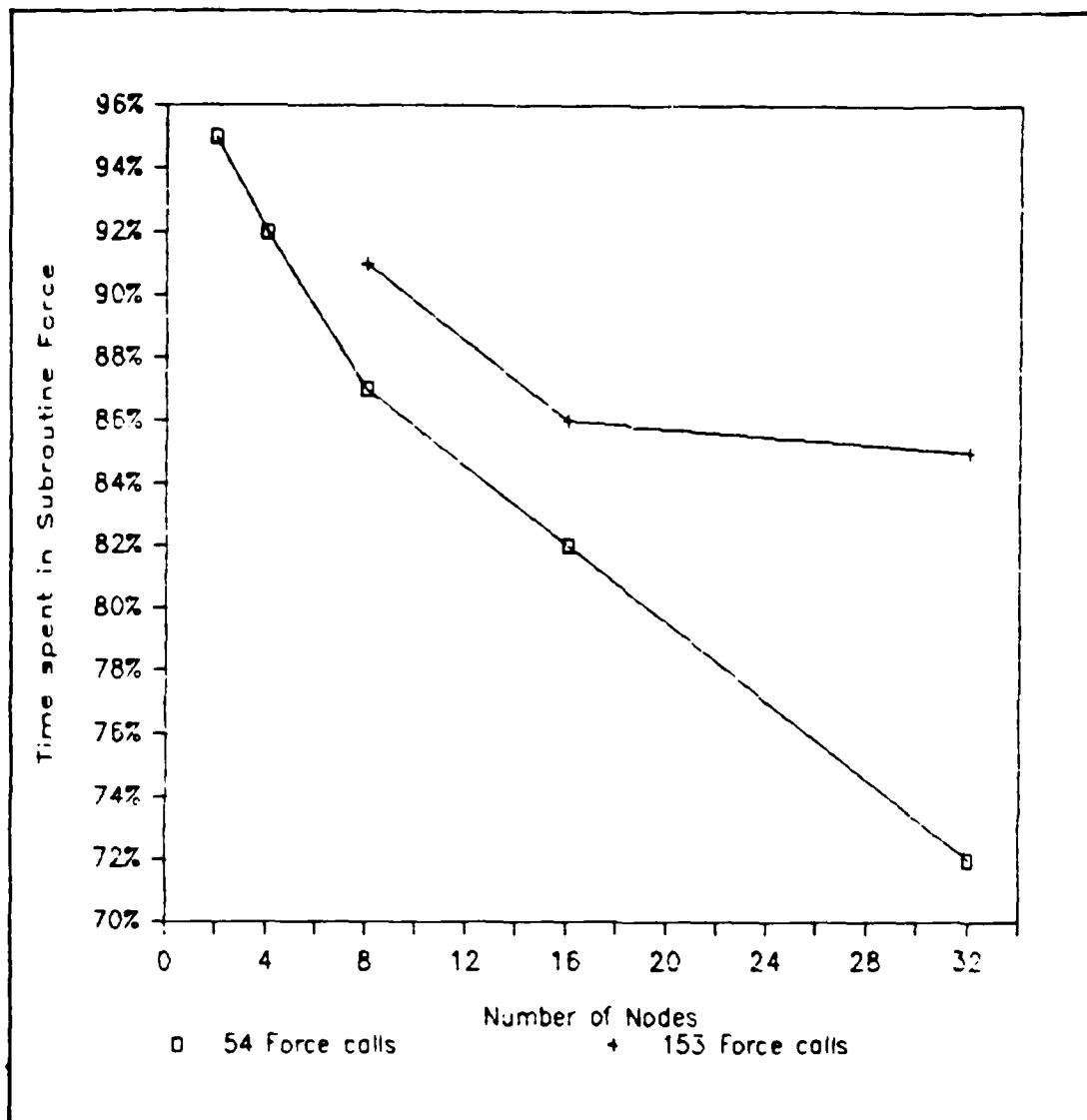


Figure 4-14. Percentage of Time in the Force and Neighbor Subroutines vs. Number of Nodes

V. CONCLUSIONS AND RECOMMENDATIONS

The overall results of this investigation are a qualified success. The Baskes and Foiles code was successfully modified for the ELIXI serial processor in that the program ran to completion and gave usable results. The computer simulated copper crystal superheated once, supercooled, melted at a temperature very near the accepted melt point, and then supercooled again. The parallel processing code showed a potential time savings by using a parallel processing computer. This chapter expands these conclusions by discussing specific aspects of the simulation process. The final section recommends what steps are needed next to improve the simulation process on a parallel computing system.

Conclusions

A central advantage of the ELIXI system is that computer jobs can be done in a delayed, or background, priority which allows a simulation to be completed overnight. Many of the experiments in this study were entered in the evening and the results were available the next morning. The iPSC parallel processor did not have this advantage.

The molecular dynamics approach using the embedded-atom method for force calculations succeeded in giving both an accurate lattice constant at room temperature and accurate linear expansions as the simulated copper crystal was heated.

Instances of superheating and supercooling not normally observed in the laboratory can be accepted because of the small size of the crystal and because of the short time periods -- two to five picoseconds -- which can be observed by the computer simulations. When the crystal was given enough time to react to outside forces, it melted or stayed solid at the accepted laboratory temperatures. In all, despite the limits of atomistic simulation, the molecular dynamics program and the embedded-atom method gave results that encourage their use in further studies.

Parallel processing, in general, still looks promising despite the researcher's inability to use all of the available time saving steps. Additional work on the program, improved programming techniques, and advanced computer hardware can all combine to make the needed changes. At this time, computer reliability and availability, problems with the parallel operating system, and the limits on host/node communications and memory sharing all degrade the utility of the iPSC as a scientific tool.

A central problem for the iPSC at the Air Force Institute of Technology is the lack of node multitasking; only one user can use the node configuration at one time. This lack, combined with the multitasking of the host processor, allows bottlenecks and delays, and means that the atomistic simulation experiments must be kept relatively short so that other operators can continue their research.

Once these problems are overcome, however, parallel processing will allow considerable advances in atomistic simulation. For example, a simulation which calculated 2300 time steps required 1 hour and 50 minutes of total wall clock time when it was run in a background mode on the ELIXI processor. The 11% time advantage of parallel processing discussed in Chapter 4 translates into a real time savings of 1 hour and 21 minutes. The computer run should only take 29 minutes in a background mode given similar delays caused by multitasking. (Calculations on the time savings are given in Appendix A.) The recommendations below should allow even greater time savings.

Recommendations

The obvious first step in the process to get the molecular dynamics problem solved on the parallel system is to modify the program so it uses all the time savings steps available on the serial processor. The neighbor list clearly does not need to be recalculated at every time step in order to get an accurate answer. Once the recalculation problem in the parallel program is solved, additional time savings can be realized.

Improved computer hardware and applications will further advance parallel processing so that it can be a scientific tool able to solve atomistic simulation problems of greater complexity. For example, the proposed change from the current Intel 80286 to Intel 80386 processors in the iPSC will shorten

processing time and increase accuracy. Communications requirements between the host and the nodes can be reduced by memory sharing, which allows all the processors to access the same memory space rather than send all information through time consuming send and receive subroutines (7:72). An operating system which allows multitasking of the nodes will greatly improve the availability of the parallel processing computer. With that improvement, several researchers will be able to run their programs simultaneously, as now possible on the serial processing computers.

As computer capability improves, the size of the crystals simulated by the computer can be increased and this advance should allow better accuracy and should avoid the problems observed in this study such as artificial superheating and supercooling. These possibilities should combine with the proven ability of molecular dynamics and the embedded-atom method to make atomistic simulation a more useful tool for the study of matter.

APPENDIX A -- CALCULATION OF THE POTENTIAL TIME SAVINGS

The purpose of this appendix is to show the calculations of the time saved by the 11% time advantage discussed in Chapters 4 and 5.

Recall that the atomistic simulation program spent 96% of its time doing the force and neighbor calculations in a serial computer and 85% of its time doing the same calculations in a parallel processing computer with 32 nodes.

A typical computer experiment of 2300 time steps ran for a total of 1 hour and 50 minutes in the background mode in the ELIXI serial processing computer (as indicated by the time at the beginning and end of a print out). That amounts to 6600 seconds (sec) of total wall clock time spent in the computer (But much less than 6600 sec of CPU time. The timing functions were not used to give total CPU time. It is assumed that multitasking beyond the control of the experimenter delayed all parts of the program equally.).

Because 4% ($100\% - 96\% = 4\%$) of the computer time is used to do all calculations beyond the force and neighbor subroutines, the total time spent doing the rest of the program is 264 sec ($6600 \text{ sec} \times 4\% = 264 \text{ sec}$) for a serial processor. With those same subroutines exported to the nodes of a parallel processing computer, the only significant changes to the computer program are in the force and neighbor subroutines; the rest of the program remains essentially unchanged and should, therefore, use the same total time of

264 seconds. The calculations not exported then take up 15% ($100\% - 85\% = 15\%$) of the total computer time. In other words, the same 264 seconds make up 15% of the computer time used by a parallel processing computer. This leads to the expectation that 1760 sec ($264 \text{ sec}/15\% = 1760 \text{ sec}$) will be the total wall clock time required for a parallel processing computer which operates at the same processor speed as the serial computer.

Thus, the 11% time advantage of using a parallel processing computer to do the force and neighbor calculations allows the parallel processor to do the entire atomistic simulation in 26.7% ($1760/6600 = 26.7\%$) of the original time for a time savings factor of 3.75 ($1/26.7\% = 3.75$).

Bibliography

1. Baskes, M.I., and S.M. Foiles. "dyn5," Computer program for Sandia National Laboratories, Livermore, California.
2. Daw, Murry S., and M.I. Baskes. "Embedded-Atom Method: Derivation and Application to Impurities, Surfaces, and other Defects in Metals," Physical Review B, 29: 6443-6453 (June 1984).
3. Daw, Murry S., and M.I. Baskes. "Semiempirical, Quantum Mechanical Calculation of Hydrogen Embrittlement in Metals," Physical Review Letters, 50: 1285-1288 (April 1983).
4. Daw, Murry S., and R.D. Hatcher. "Application of the Embedded Atom Method to Phonons in Transition Metals," Solid State Communications, 56: 697-699 (1985).
5. Edwards, John. "Parallel Processing: Wave of the Future," Online Today, 13 (September 1987).
6. Finnis, M.W., and J.E. Sinclair. "A Simple Empirical N-Body Potential for Transition Metals," Philosophical Magazine A: 50: 45-55 (1984).
7. Foiles, S.M., "Application of the Embedded-Atom Method to Liquid Transition Metals," Physical Review B, 32: 3409-3415 (September 1985).
8. Foiles, S.M., M.I. Baskes, and M.S. Daw. "Embedded-Atom-Method Functions for the FCC Metals Cu, Ag, Au, Ni, Pd, Pt, and their Alloys," Physical Review B, 33: 7983-7991 (June 1986).
9. Fox, Geoffrey C., and Paul C. Messina. "Advanced Computer Architectures," Scientific American, 67-74 (October, 1987).
10. Haile, J.M., Associate Professor of Chemical Engineering. A Primer on the Computer Simulation of Atomic Fluids by Molecular Dynamics. Clemson University, 1980.
11. Haliday, David, and Robert Resnick. Fundamentals of Physics. New York: John Wiley & Sons, 1974.
12. McKelvey, John P., Solid State and Semiconductor Physics. Malabar FL: Robert E. Krieger Press, 1986.

13. Metzler, William D. Evaluation of Tersoff's Revised Silicon Potential in a Molecular Dynamic Simulation of Solid Silicon. MS Thesis, AFIT/GNE/ENP/87M-5. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 1987.
14. Reif, F. Fundamentals of Statistical and Thermal Physics. New York: McGraw-Hill, 1965.
15. Sabochick, Michael., Assistant Professor of Nuclear Engineering. Basic Primer on Atomistic Simulation. Air Force Institute of Technology.
16. "Technologies for the Eighties," Science and Technology, edited by Richard H. Stephens and Eston T. White. Washington D.C.: National Defense University, 1983.
17. Thee, Paul L. Calculation of the Equation of State for Silicon. MS Thesis, AFIT/GEP/ENP/86D-10. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1986.
18. Touloukinn, Y.S., and others. Thermophysical Properties of Matter, Volume 12: Thermal Expansion, Metallic Elements and Alloys. New York: IFI/Plenum, 1977.
19. Yeo, Yung Kee. Class handout distributed in Physics 7.70, Solid State Physics I, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, October 1986.

VITA

Major Charles M. Court was born on 5 March 1949 in Oak Park, Illinois. He graduated from the University of Illinois at Urbana in 1971 with Bachelor degrees in both Physics and Political Science. After graduation, he was commissioned at the Officer's Training School, Medina Air Force Base, Texas. Major Court holds the aeronautical rating of senior navigator, and has served as an instructor electronic warfare officer in the F-105G and F-4G Wild Weasel weapons systems. His prior assignments include flying duty with the 561st and the 563rd Tactical Fighter Squadrons, George AFB, California; the 81st Tactical Fighter Squadron, Spangdahlem AB, Federal Republic of Germany; and staff work as a Test Realism Manager, Air Force Operational Test and Evaluation Center, Kirtland AFB, New Mexico. Major Court also holds a Master of Business Administration degree from Golden Gate University and is a graduate of the National Defense University, Washington D.C. He entered the School of Engineering, Air Force Institute of Technology in June, 1986.

Permanent address: 109 Princeton

Des Plaines, Illinois 60016

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b. RESTRICTIVE MARKINGS		
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION / AVAILABILITY OF REPORT		
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE			Approved for public release; distribution unlimited		
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GEP/ENP/87D-4			5. MONITORING ORGANIZATION REPORT NUMBER(S)		
6a. NAME OF PERFORMING ORGANIZATION School of Engineering		6b. OFFICE SYMBOL (If applicable) AFIT/ENP		7a. NAME OF MONITORING ORGANIZATION	
6c. ADDRESS (City, State, and ZIP Code) Air Force Institute of Technology Wright-Patterson AFB, Ohio 45433				7b. ADDRESS (City, State, and ZIP Code)	
8a. NAME OF FUNDING / SPONSORING ORGANIZATION		8b. OFFICE SYMBOL (If applicable)		9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c. ADDRESS (City, State, and ZIP Code)				10. SOURCE OF FUNDING NUMBERS	
PROGRAM ELEMENT NO.		PROJECT NO.		TASK NO.	WORK UNIT ACCESSION NO.
11. TITLE (Include Security Classification) A STUDY OF THE EQUATION OF STATE FOR COPPER					
12. PERSONAL AUTHOR(S) Charles M. Court, B.S., B.A., M.B.A., Major, USAF					
13a. TYPE OF REPORT M.S. Thesis		13b. TIME COVERED FROM _____ TO _____		14. DATE OF REPORT (Year, Month, Day) 1987 December 7	
				15. PAGE COUNT 71	
16. SUPPLEMENTARY NOTATION					
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUB-GROUP			
20	02		Molecular Dynamics, Copper		
11	06	01			
19. ABSTRACT (Continue on reverse if necessary and identify by block number)					
Thesis Chairman: Michael J. Sabochick, Captain, USAF Assistant Professor of Nuclear Engineering Air Force Institute of Technology (AFIT/ENP)					
Approved for public release: LAW AFR 190-17 <i>[Signature]</i> 24 Feb 88 AFIT/ENP Development Wright-Patterson AFB, OH 45433					
20. DISTRIBUTION / AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED		
22a. NAME OF RESPONSIBLE INDIVIDUAL Michael J. Sabochick, Captain, USAF			22b. TELEPHONE (Include Area Code) 513-255-2012		22c. OFFICE SYMBOL AFIT/ENP

UNCLASSIFIED

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

The equation of state of copper was studied by using computer simulations. An existing molecular dynamics program written for the Cray was modified to run on both the VAX and on a parallel processing system. The program applied the embedded-atom method to determine interatomic forces.

Experiments on the VAX found the copper melting point to be very near 1350° K which is close to the known melt point of 1357.6° K.

Measurements taken on both types of computers showed that parallel processing offers a time savings factor of 3.75 over serial processing systems.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

END
DATE
FILMED
MARCH
1988
DTIC